National Imagery Transmission Format Standard

TUTORIAL

Prepared by:
Joint Interoperability Test Command

Updated:
18 October 2001

TUTORIAL OUTLINE

A. BACKGROUND
B. FORMAT OVERVIEW
C. DIGITAL IMAGERY CONCEPTS (Overview)
D. FORMAT DETAILS
E. COMPRESSION
F. ANNOTATION
G. EXTENSIONS
H. PRODUCTS
I. WRAP-UP

BACKGROUND

- NITFS BACKGROUND
- STANDARDS COMMITTEES
- NITFS EVOLUTION STRATEGY
- NITFS OBJECTIVES
- NITFS OPERATIONAL CONCEPT
- NITFS SUITE OF STANDARDS
- RELATED STANDARDS EFFORTS
- NITFS TEST AND EVALUATION PROGRAM

BACKGROUND

- 1989 - NITF 1.1 published and disseminated for general use (March)
- 1990 - NITF Certification Test Facility (CTE) established
- 1991 - NITF began conversion to Department of Defense Standards, name changes to NITFS
- 1993 - Continued development and validation of NITF 2.0
- 1994 - Certification testing of NITF 2.0 systems & software
- 1995 - Began development of: MIL-STD-2500B , ISO 12087-5 BIIF, STANAG 4545 NSIF
- 1998 - ISO 12087 BIIF approved as an international standard (Feb.)
  - NITF 2.1 (2500B) certification testing begins (Oct.)
  - STANAG 4545 NSIF ratified 2 December

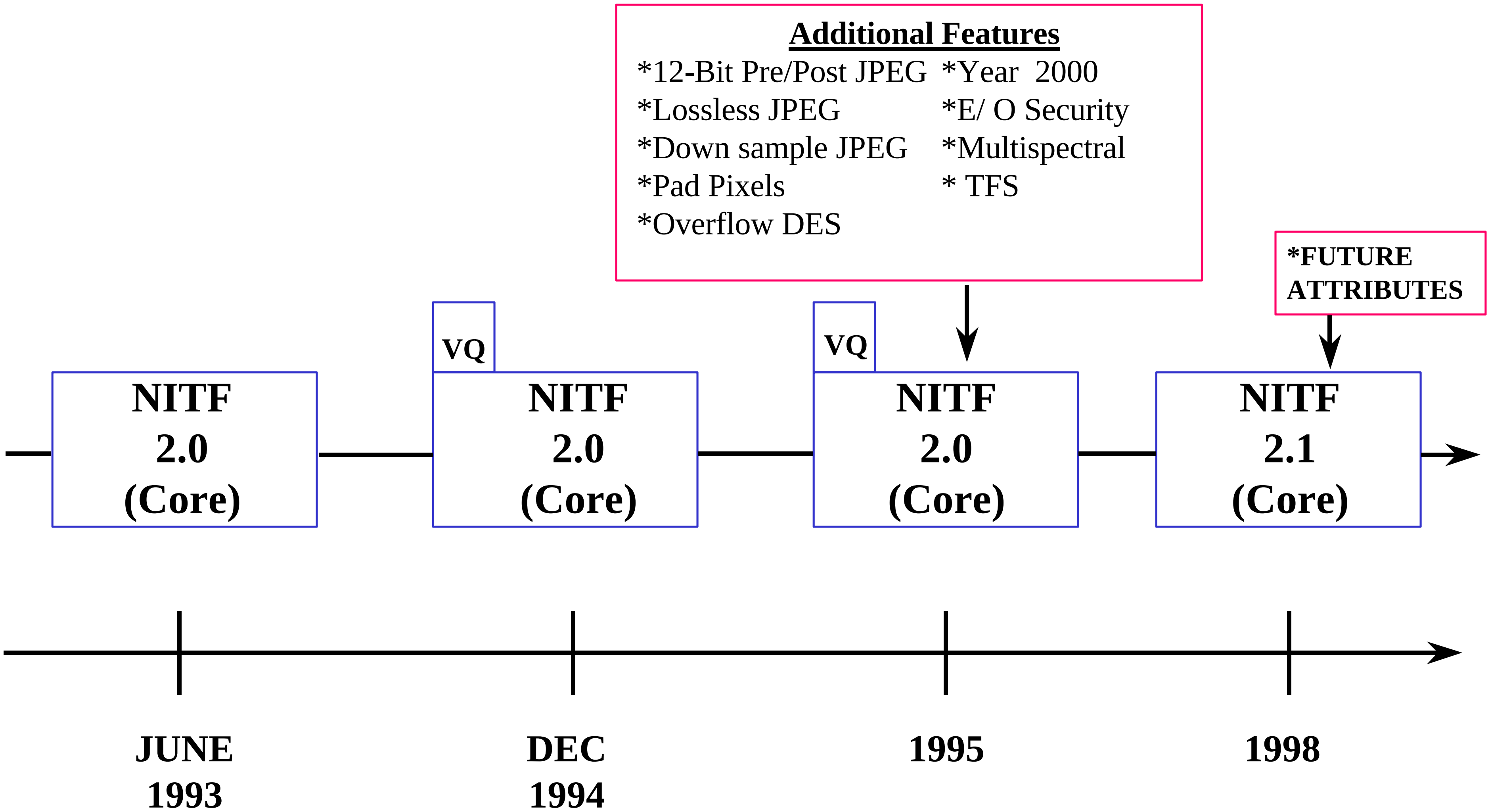
DEFENSE STANDARDIZATION PROGRAM

Standards Coordinating Committee (SCC)

- **Geospatial Standards Management Committee (GSMC)**
- **Imagery Standards Management Committee (ISMC)**
  - NITFS Technical Board (NTB)
    - Format Working Group (FWG)
    - Bandwidth Compression Working Group (BWCWG)
    - Additional adhoc Working Groups as needed
  - Motion Imagery Working Group (MIWG)
  - CIGSS Standards Working Group (CSWG)

NIMA/IPS (Standards) has the responsibility for the NITFS

NITF EVOLUTION STRATEGY



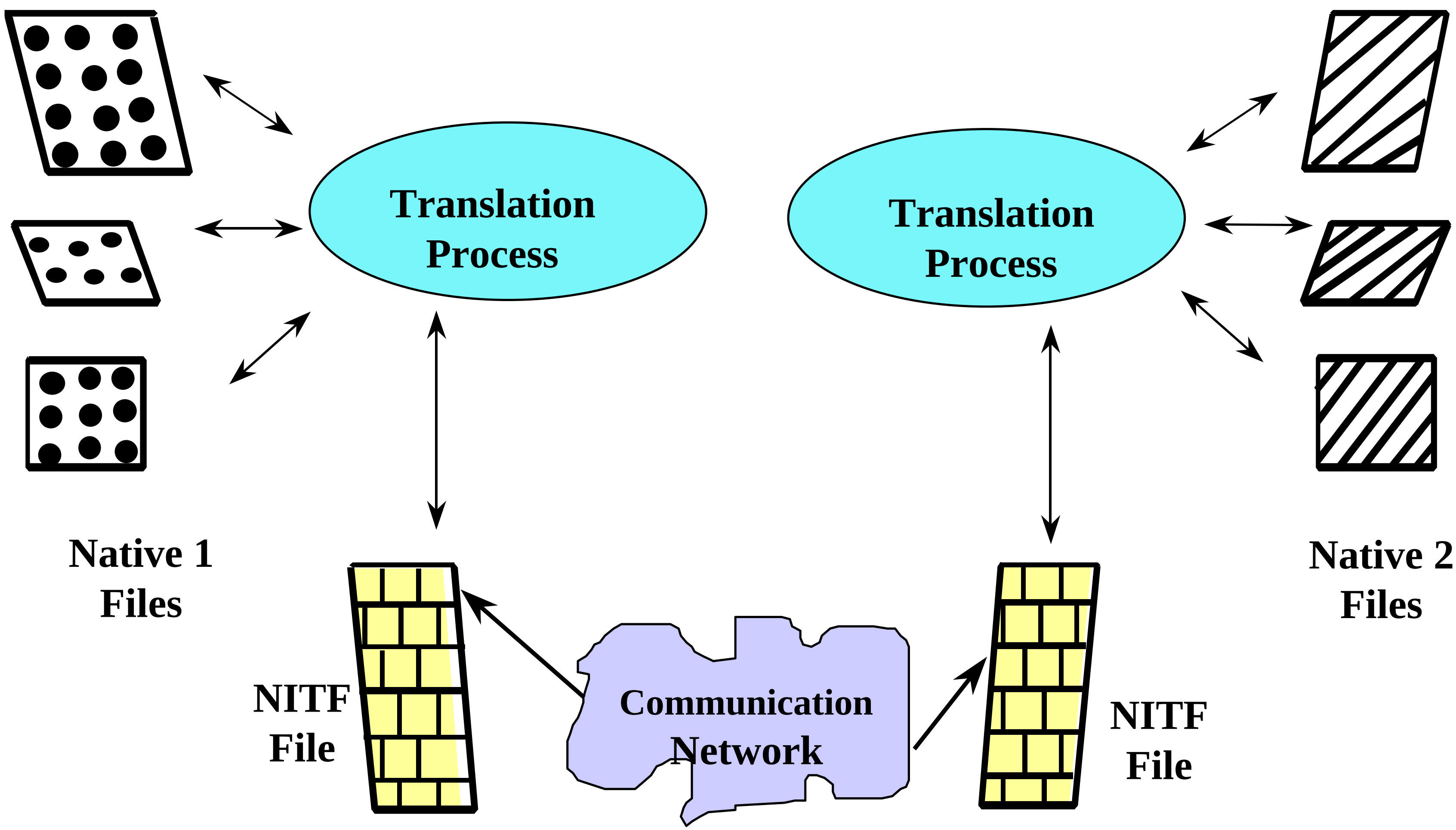
NITFS OBJECTIVES

- **To provide a means for diverse systems to share imagery & associated data.**
- **To enable systems to exchange comprehensive information in a single file to users with diverse needs & capabilities.**
- **To enable users to select only data items that correspond with their needs & capabilities.**
- **To minimize the cost & schedule required to achieve the capability to share imagery & associated data.**

NITFS OPERATIONAL CONCEPT

- **Data interchange between systems is potentially enabled by a cross-translation process.**
- **Systems only need to comply with one (vice many) external formats.**
- **Systems will have the capability to translate internal representation for data to and from the NITF file format.**
- **Systems will have the capability to exchange with one or more recipients using: Tactical Communication Protocol (TACO2), TCP/ IP, LAN/WAN, Command WANs, and/or Media (e.g., tape, CD, etc.).**

NATIVE MODE TO NITF



NITF 2.0 CORE SUITE OF STANDARDS

- **MIL-STD-2500A** National Imagery Transmission Format (NITF) (Version 2.0) for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-2301** Computer Graphics Metafile for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-196** Bi-Level Image Compression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-197A** Adaptive Recursive Interpolative Differential Pulse Code Modulation (ARIDPCM) Image Compression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-198A** Joint Photographic Experts Group (JPEG) Image Compression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-199** Vector Quantization Decompression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-2045-44500** Tactical Communications Protocol 2 (TACO2) for the National Imagery Transmission Format Standard (NITFS).

NITF 2.1 CORE SUITE OF STANDARDS

- **MIL-STD-2500B** National Imagery Transmission Format (NITF) (Version 2.1) for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-2301A** Computer Graphics Metafile for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-196** Bi-Level Image Compression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-198A** Joint Photographic Experts Group (JPEG) Image Compression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-188-199** Vector Quantization Decompression for the National Imagery Transmission Format Standard (NITFS).
- **MIL-STD-2045-44500** Tactical Communications Protocol 2 (TACO2) for the National Imagery Transmission Format Standard (NITFS).

http://164.214.2.51/ntb/

ADDITIONAL NITFS KEY DOCUMENTS

- **N-0105/98 -** National Imagery Transmission Format Standard (NITFS) Standards Compliance and Interoperability Test and Evaluation Program Plan Version 1.0
- **N-0106/98 -** National Imagery Transmission Format Standard (NITFS) Bandwidth Compression Standards and Guidelines
- **STDI-0002** - The Compendium of Controlled Extensions (CE) for the NITFS

http://164.214.2.51/ntb/

RELATED STANDARDS EFFORTS

- **ISO/IEC 12087-5 Basic Image Interchange Format (BIIF)**
- **STANAG 4545 NATO Secondary Imagery Format (NSIF)**
- **Digital Geographic Information Exchange Standard (DIGEST)**
- **Open Skies Treaty (BIIF Profile)**
- **Spatial Data Transfer Standard (SDTS), Part 5 (Raster Profile)**
- **Commercial Satellite Data Product Specifications**
- **Airborne Data Product Specifications**
- **UAV/Reconnaissance Data Product Specifications**

NITFS TEST PROGRAM

- **Test Director**
  - **Mr. Stephen Kerr**
- **Sponsor**
  - **National Imagery and Mapping Agency (NIMA)**
- **Objectives**
  - **Executive Agent for NIMA Imagery-related Testing**
  - **Standards Validation Testing**
  - **Standards Conformance Testing**
  - **Standards Based Interoperability Testing**
- **Clients:**
  - **DoD & Federal Agencies**
  - **NATO**
  - **Commercial Vendors**

STANDARDS VALIDATION TESTING

- Is the proposed standard/specification:
  - Complete?
  - Technically Accurate?
  - Free of Conflicts?
  - Unambiguous?
  - Testable for Conformance?

- Does a realization of the proposed specification do what was intended?

- Establish the criteria for determining if the specification has been implemented adequately.

VALIDATION METHODOLOGY

Conceptually a five step process:

1 - Identify the service, functional, and performance requirements

2 - Develop proposed conformance test objectives, test criteria, and test cases.

3 - Develop a Sample Implementation(s) and the Means of Test (MOT).

4 - Test the sample until the specification, sample, and MOT are in harmony.

5 - Evaluate the degree to which the sample meets service, functional, and performance objectives.

FREQUENTLY ASKED QUESTIONS….

Q. What exactly is NITFS Conformance Testing?

A. Conformance testing is a methodology that ensures all vendors and developers of imagery products interpret the NITF Standards in a uniform manner, thereby reducing the possibility of interoperability problems during system/product fielding.

Q. What does conformance testing reveal that can't be discovered by developmental or other testing performed in my own facility?

A. The NITFS Conformance Testing Program is universal. Vendors and developers around the world participate in the program. It would be impossible to expect every vendor or developer to interpret every specification in the same manner. Accordingly, the interpretations provided by NITFS CTE personnel provide the consistency and uniformity needed for improved interoperability capabilities.

MORE FREQUENTLY ASKED QUESTIONS...

Q. How much can I expect to pay for the JITC's NITFS Conformance Testing services?

A. Exact costs for the JITC's services are test dependent and are beyond the scope of this document; however, a reasonably accurate "rule of thumb" estimation can be based on a charge of approximately $1800 per test day.

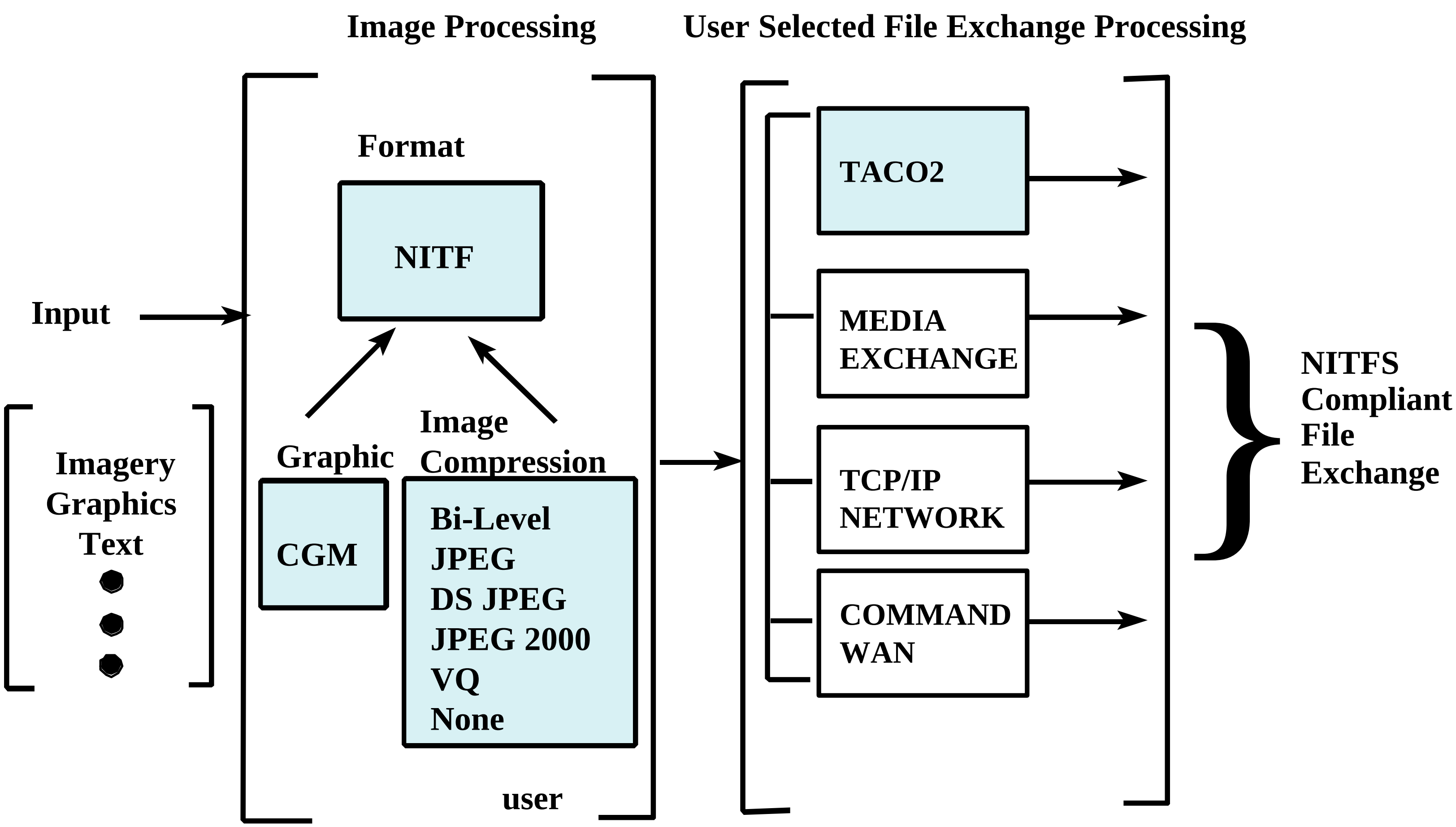
Q. How far in advance must I schedule a conformance test with the JITC?

A. Clients may call at anytime to discuss and schedule a test; however, the completed test forms and payment must be received prior to the test start date. Test periods are scheduled on a first-come, first-served basis, with exceptions and considerations available for bonafide emergencies or other unusual circumstances. Testing can usually be scheduled to begin within 6-10 weeks of a bonafide request for test services.

FORMAT OVERVIEW

- **NITF FILE STRUCTURE**
  - **Fixed Fields**
  - **Extensible Fields**
  - **Extensible Segments**
- **VARIETY OF DATA SEGMENT TYPES**
  - **Images**
  - **Graphics (Symbols)**
  - **Labels (Phased out)**
  - **Text**
  - **Data Extension Segments**
  - **Reserved Extension Segments**

NITF COMPONENT RELATIONSHIPS



NITF FILE STRUCTURE

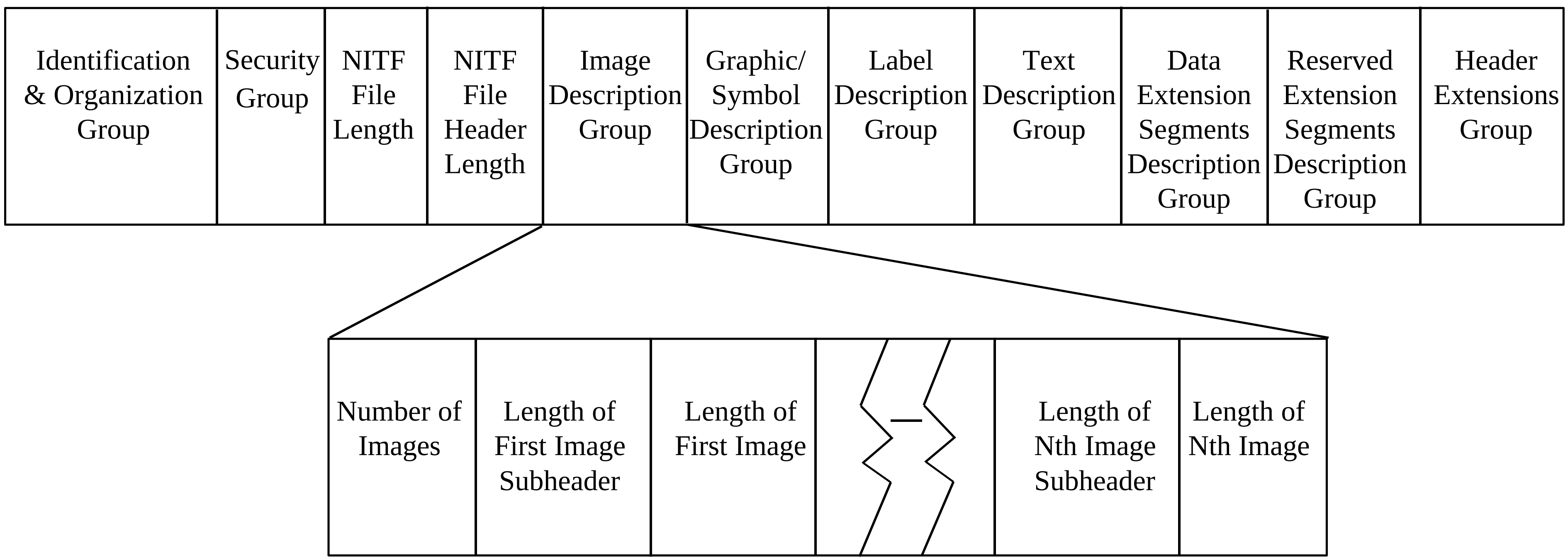
NITF 2.0 and NITF 1.1

NITF File Header	Image Segments	Symbol Segments	Label Segments	Text Segments	Data Extension Segments	Reserved Extension Segments

NITF 2.1

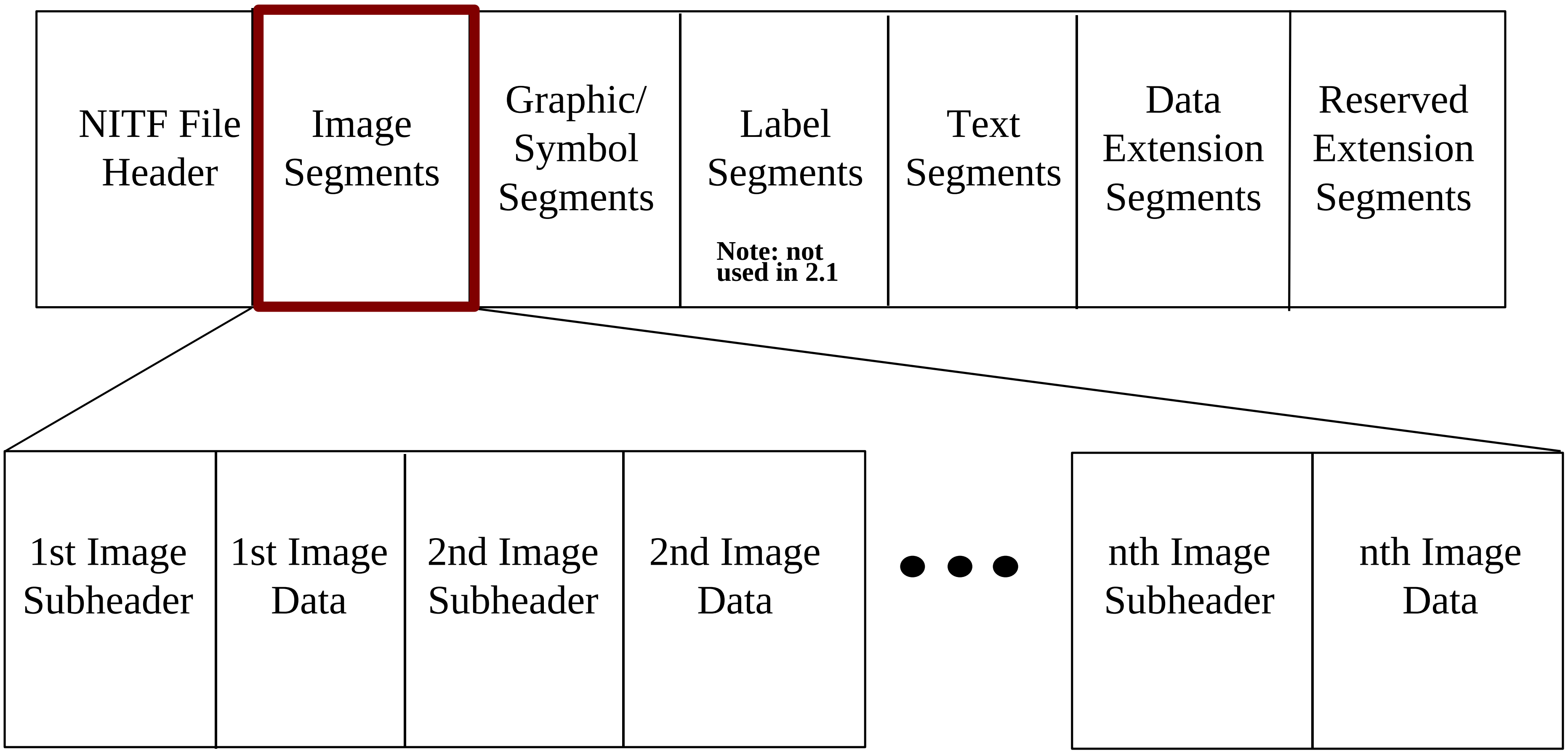
NITF File Header	Image Segments	Graphic Segments	Text Segments	Data Extension Segments	Reserved Extension Segments

NITF FILE HEADER STRUCTURE

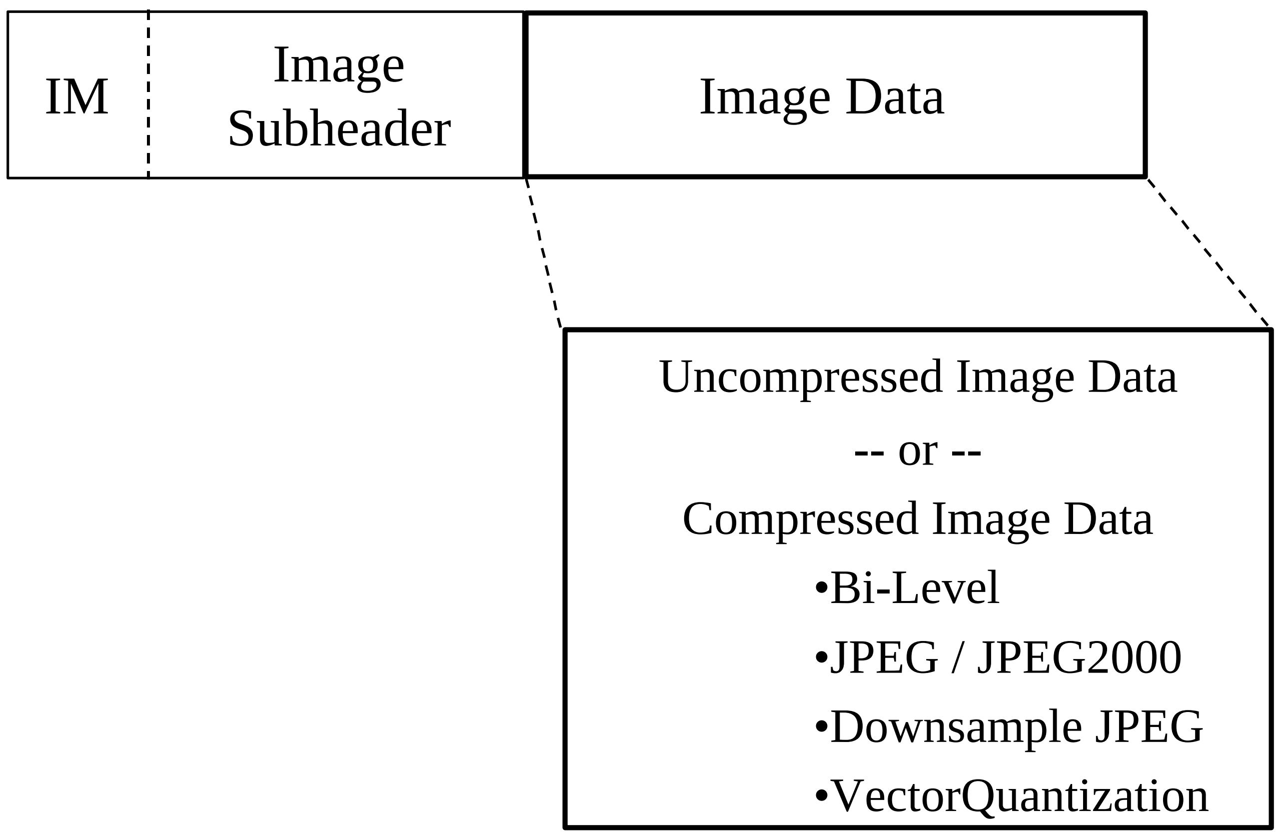


NOTE: Date/Time fields changed in NITF 2.1 for Y2K
Security Group fields changed in NITF 2.1 to comply with EO 12958

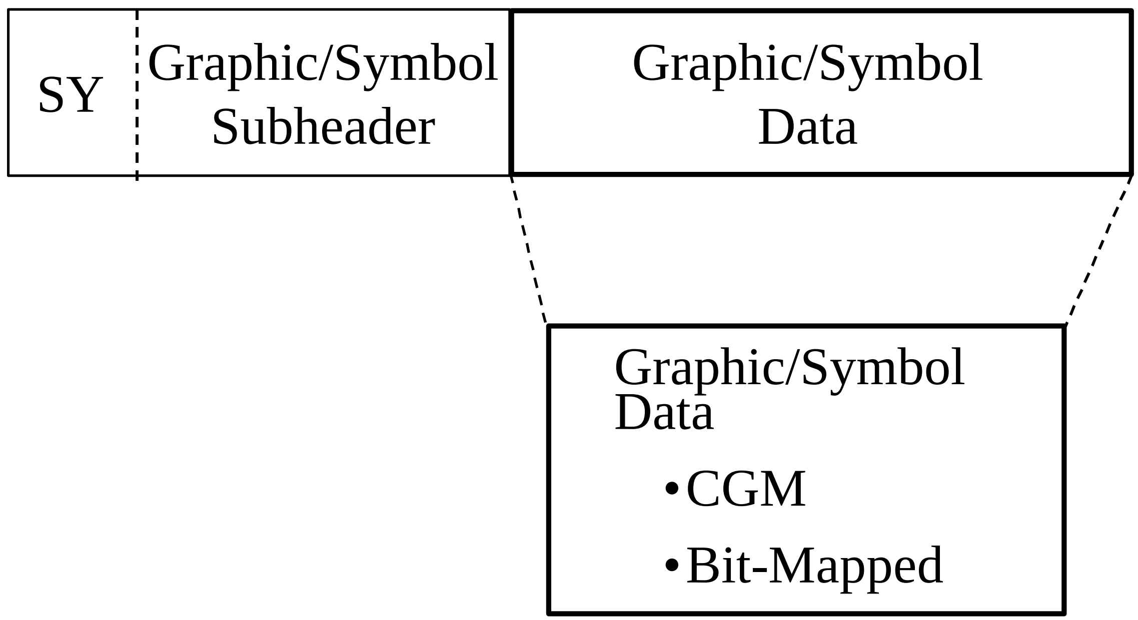
NITF IMAGE SEGMENT FILE STRUCTURE



NITF IMAGE SEGMENT FILE STRUCTURE (CONT'D)

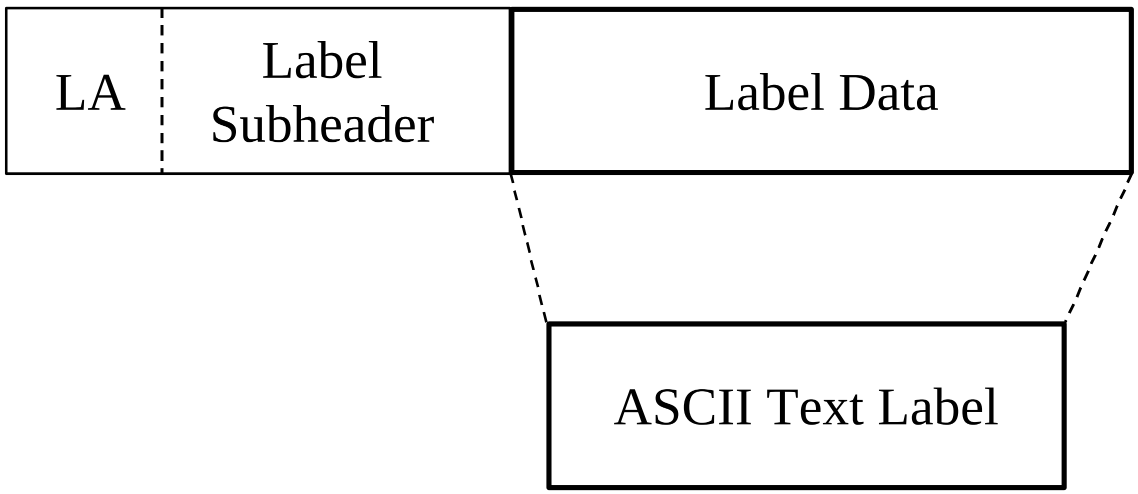


NITF SYMBOL SEGMENT FILE STRUCTURE



Note: Bit-Mapped symbols are not used in NITF 2.1

NITF LABEL SEGMENT FILE STRUCTURE



Note: Label Segments are not used in NITF 2.1

NITF TEXT SEGMENT FILE STRUCTURE

TE	Text Subheader	Text Data

- STA - Basic Latin Characters
- UT1 - Basic Latin and Latin Supplement 1 Characters, 1-byte encoded
- MTF - Message Text Format (MIL-STD-6040)
- U8S - Basic Latin and LS-1 UTF-8 encoded

NITF DATA/RESERVED EXTENSION SEGMENT (DES/RES) FILE STRUCTURE

DE RE	DES/RES Subheader	Extension Data

(NITF 2.0)

DES Data

- Registered Extensions
- Controlled Extensions
- STREAMING_FILE_HEADER

RES Data

- None Defined

(NITF 2.1)

DES Data

- TRE_ OVERFLOW
- STREAMING_FILE_HEADER

RES Data

- None Defined

DIGITAL IMAGERY CONCEPTS

NITFS Vocabulary:

- **Rasters, Pixels, Etc.**
- **Blocking, Masking, and Interleaving**
- **Color Look-Up Tables**
- **Imagery Annotations**
- **Textual Adjuncts**
- **Common Coordinate System**
- **Display and Attachment Levels**
- **Support Data Extensions**

NITFS VOCABULARY

- **RASTER**
- **PIXELS**
- **BLOCK**
- **PAD PIXELS**
- **IMODE**
- **MASKED BLOCKS**
- **BANDS**
- **LUT**

Raster: Images defined as a set of pixels or dot values in row (line) and column (sample) format. A two-dimensional array of pixel values.

Pixel: The atomic element of an image having a discreet value. Although a pixel value represents a minute area of an image, the generic use of the term does not specify the exact shape or symmetry of the area (circle, oval, square, rectangle, other) represented by the value. An acronym for a Picture Element.

Block: A rectangular array of pixels. An image consists of the union of one or more non-overlapping blocks. (Synonymous with tile).

Pad Pixels: 1) A fill pixel given a specific identification value within an image block. Pad pixels are included to ensure that each block is filled with contiguous pixel values, but should be interpreted as having no meaning. 2) Transparent Pixel is a specified pad pixel value which must be interpreted to not obscure any underlying pixel value over which it may be located.

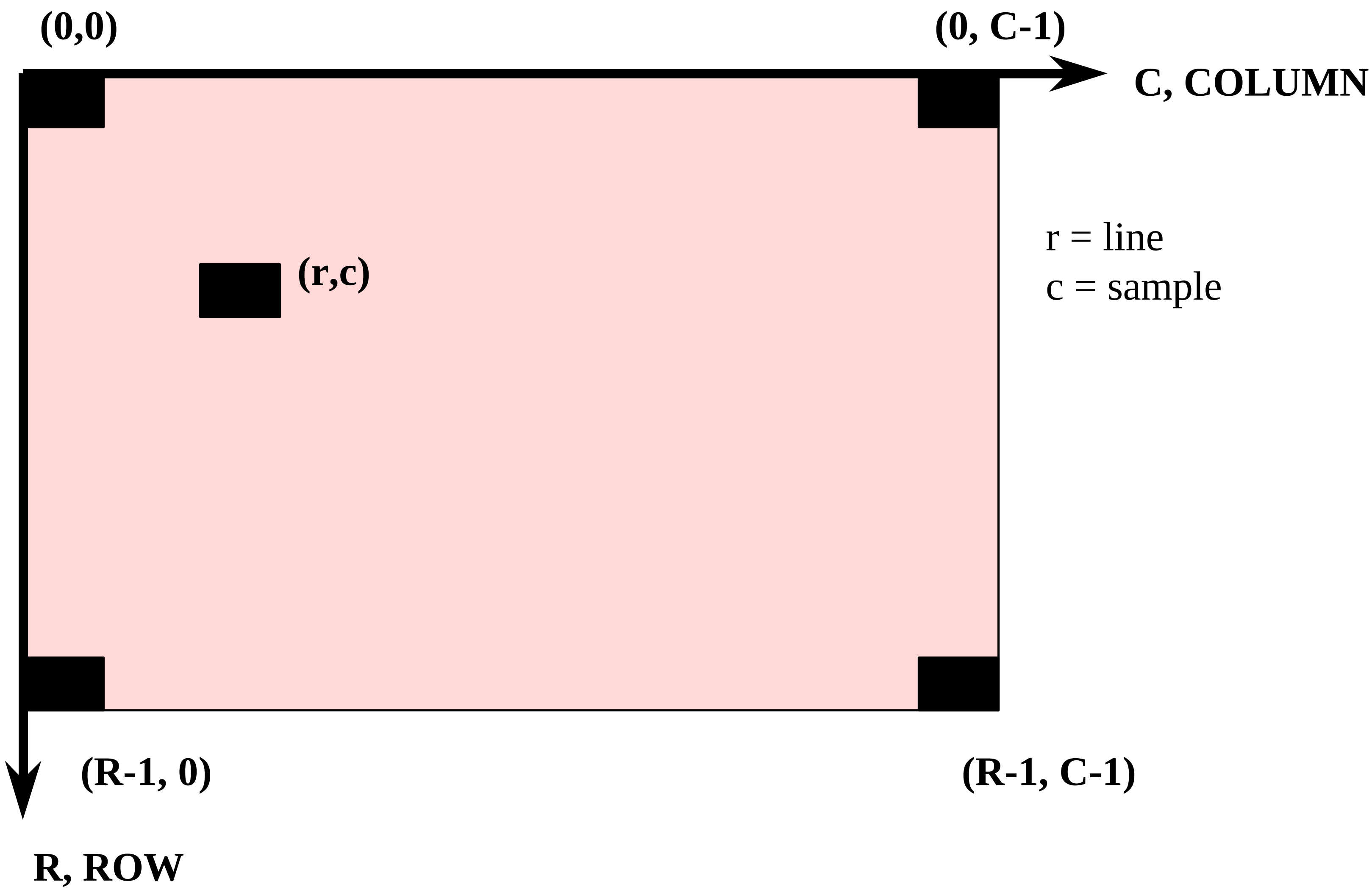
IMODE: A field in the NITF image subheader used to indicate whether the image bands are positioned sequentially or interleaved (by block or pixel).

Masked Blocks: A structure which identifies the blocks in a blocked (tiled) image which contain no valid data, and which are not transmitted or recorded. The structure allows the receiver to recognize the offset for each recorded/transmitted block. For example, a 2x2 blocked image which contained no valid data in the second block (block 1) would be recorded in the order: block 0, block 2, block 3. The blocked image mask would identify block 1 as a non-recorded/non-transmitted block, and would allow the receiving application to construct the image in the correct order.

Band: One of the two-dimensional (row/column) pixel value arrays that comprise an image. In the case of gray scale, bi-level, or 8-bit color images, the representation is a single two-dimensional array; in the case of 24-bit true color images (8 bits for each of three color elements), the representation is three two-dimensional array of pixel values.

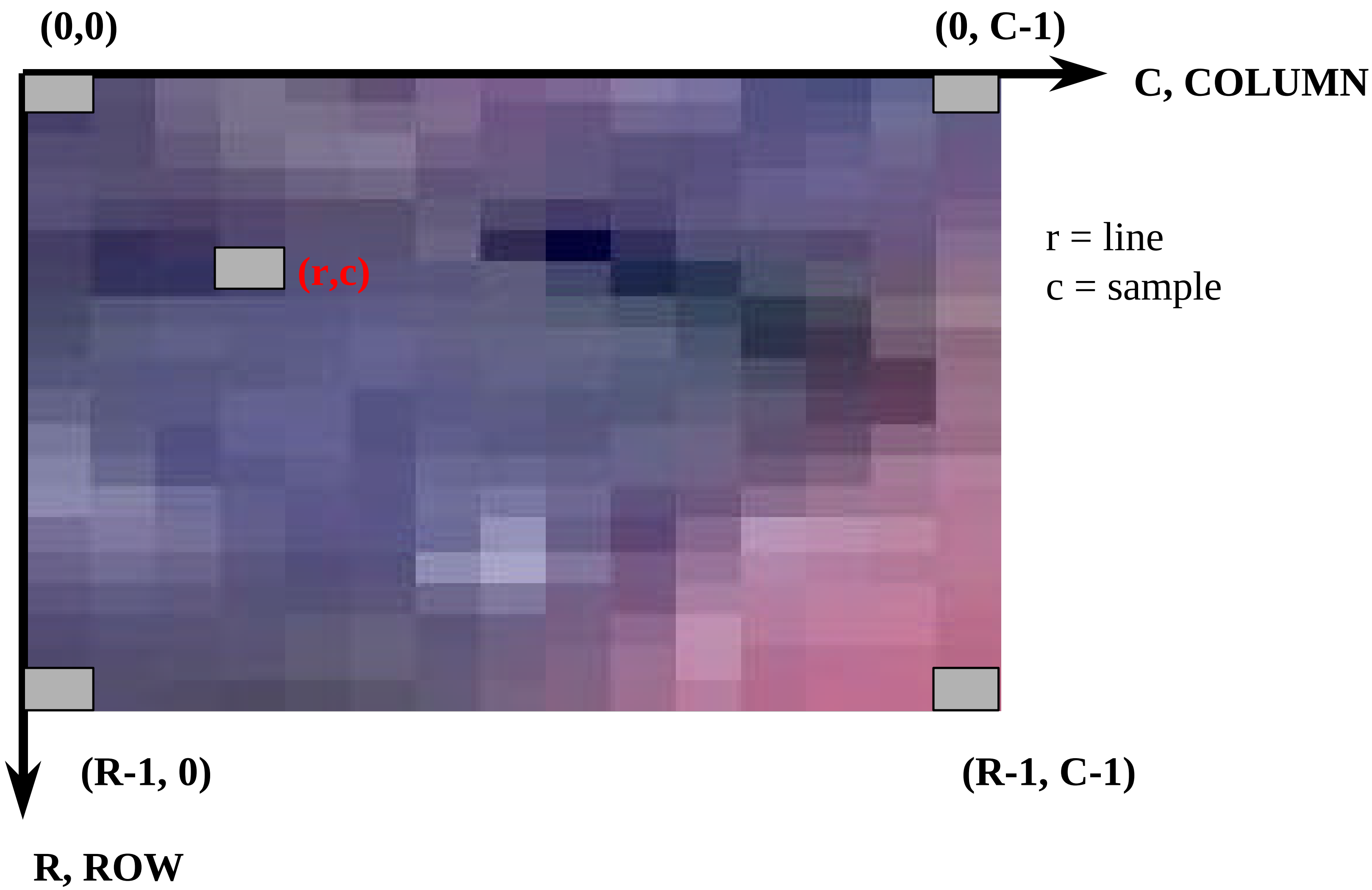
Look-up table (LUT): A means to translate a set of input pixel values into an alternate set of output pixel values. For example, 8-bit pseudo color pixel values may be used as an index into a LUT that has 256 entries of true color (24-bit) pixel values.

RASTER/PIXELS



r = line
c = sample

RASTER/PIXELS



PIXEL/DATA VALUE TYPES

Bi-Level
- One-bit value
- Represents two colors or shades of gray

Integer
- N-bits value
- Represents 2^N colors or shades of gray
  - For N = 8, 256 shades of gray
  - For N = 12, 4096 shades of gray

Signed Integer
- N-bits value
- Positive and negative values (e.g., Elevation Postings)

Real
- IEEE 32/64-bit value
- Floating Point data representation

Complex
- IEEE 64-bit value
- 32-bit real value
- 32-bit imaginary value

BANDS

Single Band

One pixel value per raster array location.
(e.g., one 8-bit integer allows 256 shades of gray)

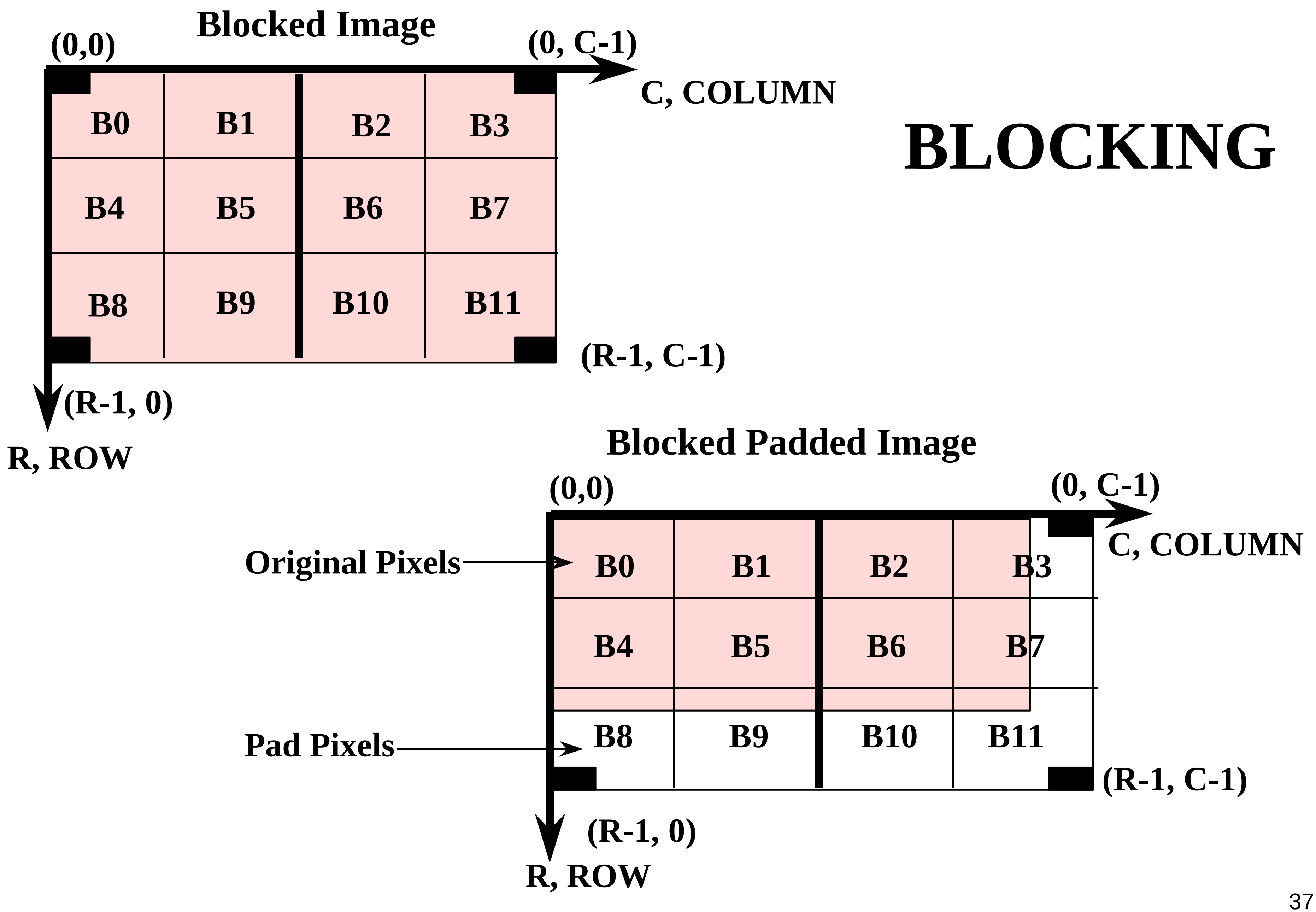
Multiple Band

N pixel values per raster array location.

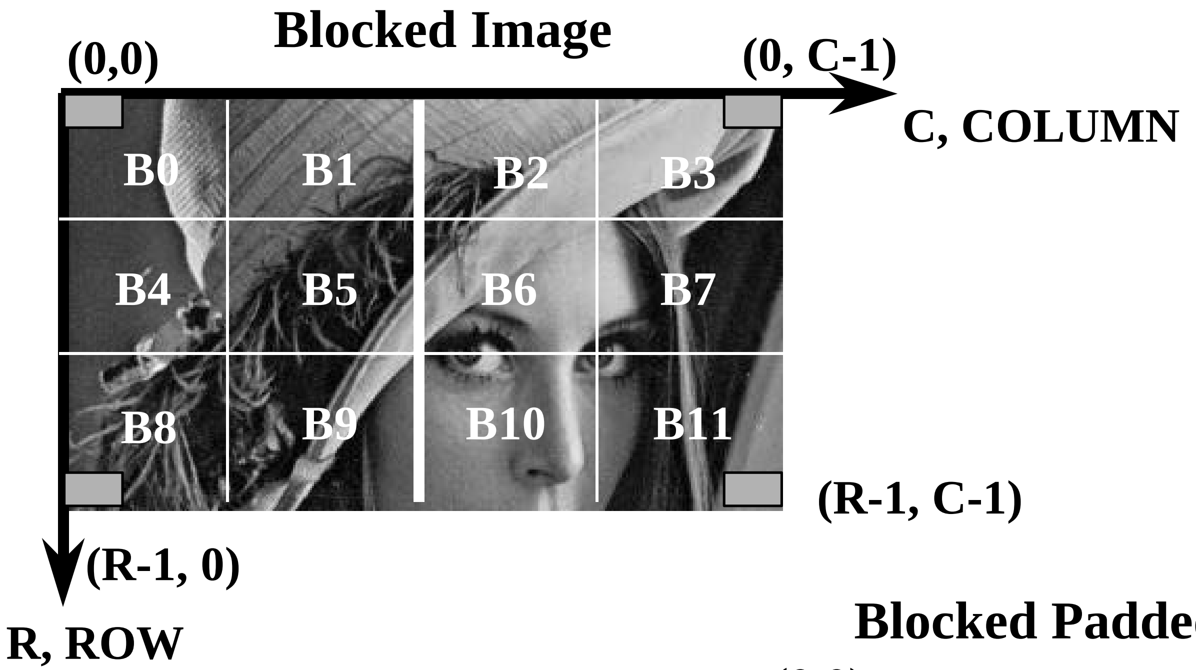
Examples:

- RGB   Three 8-bit integer values per pixel
- YCbCr Three 8-bit integer values per pixel
- Multi (e.g., seven 8-bit integer values per pixel)
- Multi (e.g., 256 32-bit real values per pixel)

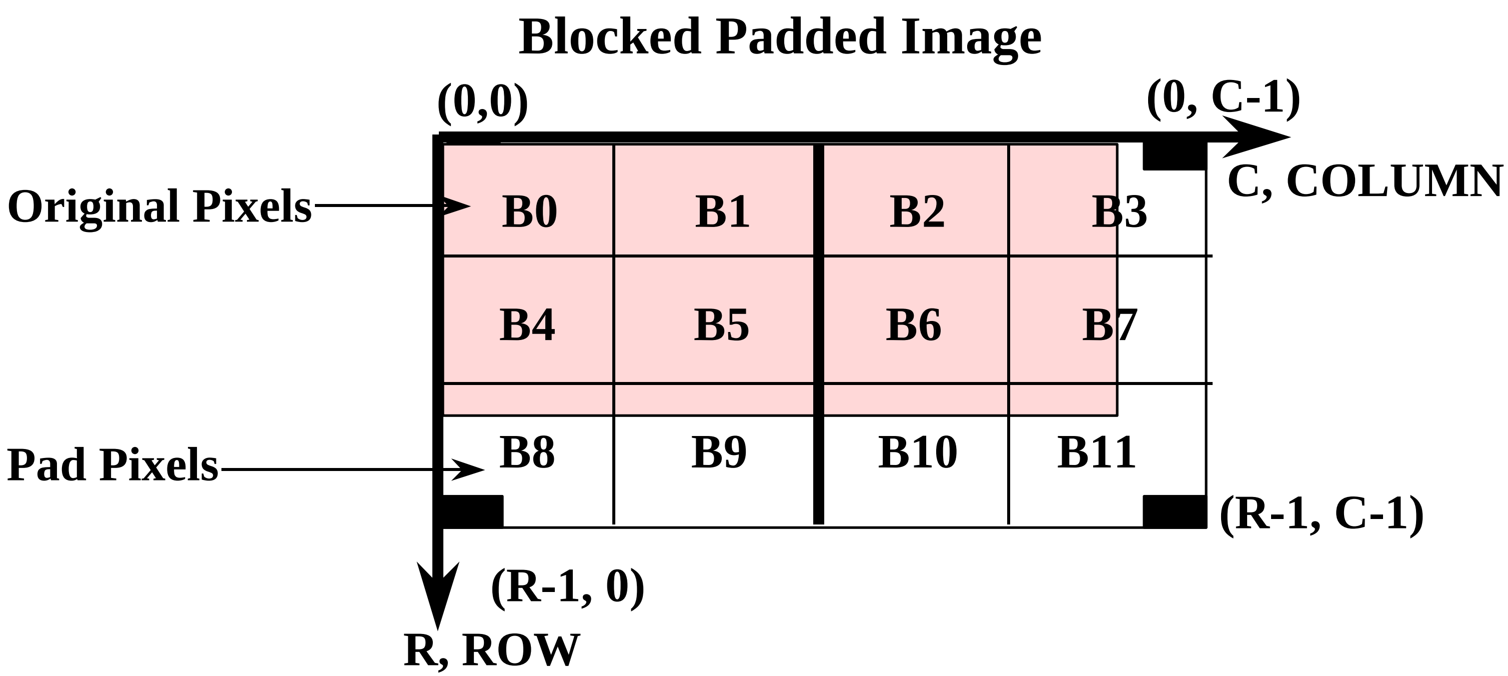
BLOCKING


Blocked Image
(0,0)
(0, C-1)
C, COLUMN
B0
B1
B2
B3
B4
B5
B6
B7
B8
B9
B10
B11
(R-1, C-1)
(R-1, 0)
R, ROW
Blocked Padded Image
(0,0)
(0, C-1)
C, COLUMN
Original Pixels
B0
B1
B2
B3
B4
B5
B6
B7
Pad Pixels
B8
B9
B10
B11
(R-1, C-1)
(R-1, 0)
R, ROW

BLOCKING

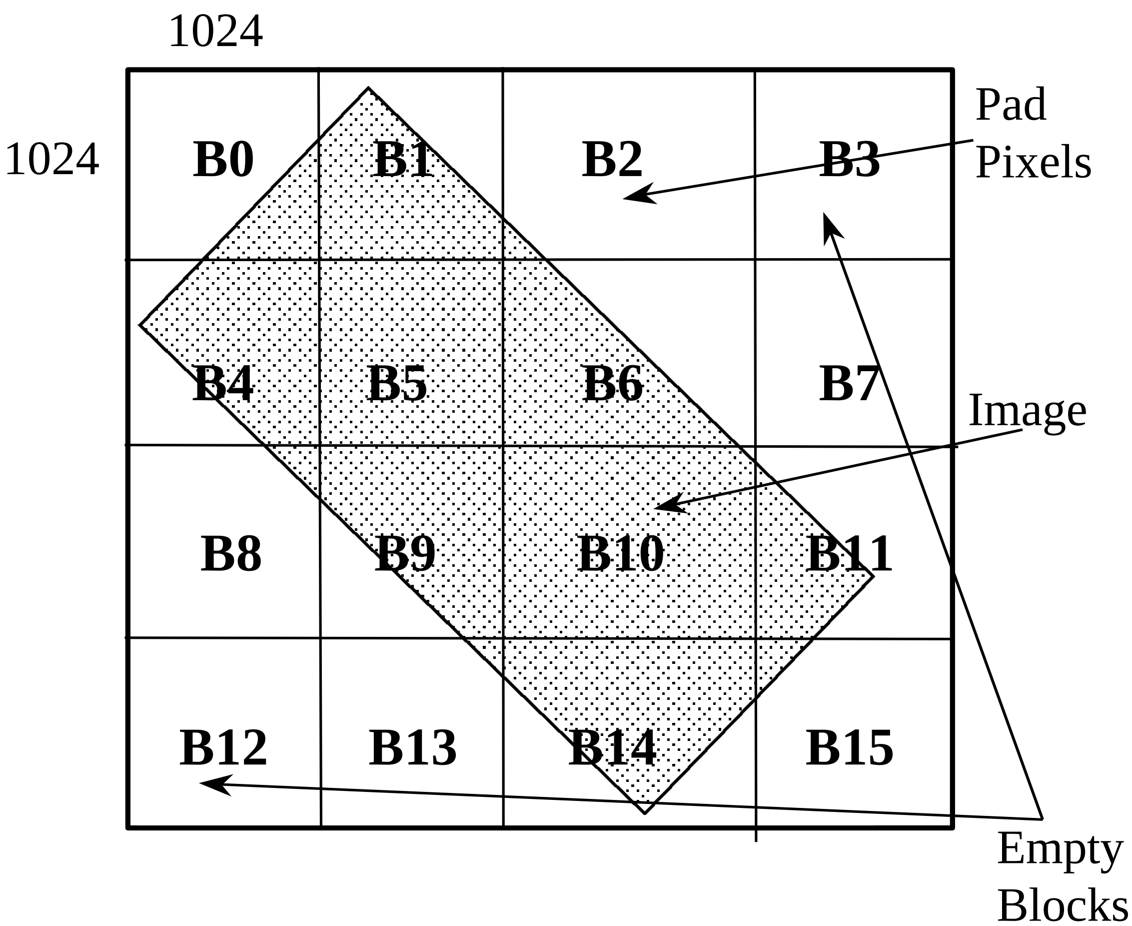


Blocked Image
(0,0)
(0, C-1)
C, COLUMN
B0
B1
B2
B3
B4
B5
B6
B7
B8
B9
B10
B11
(R-1, C-1)
(R-1, 0)
R, ROW



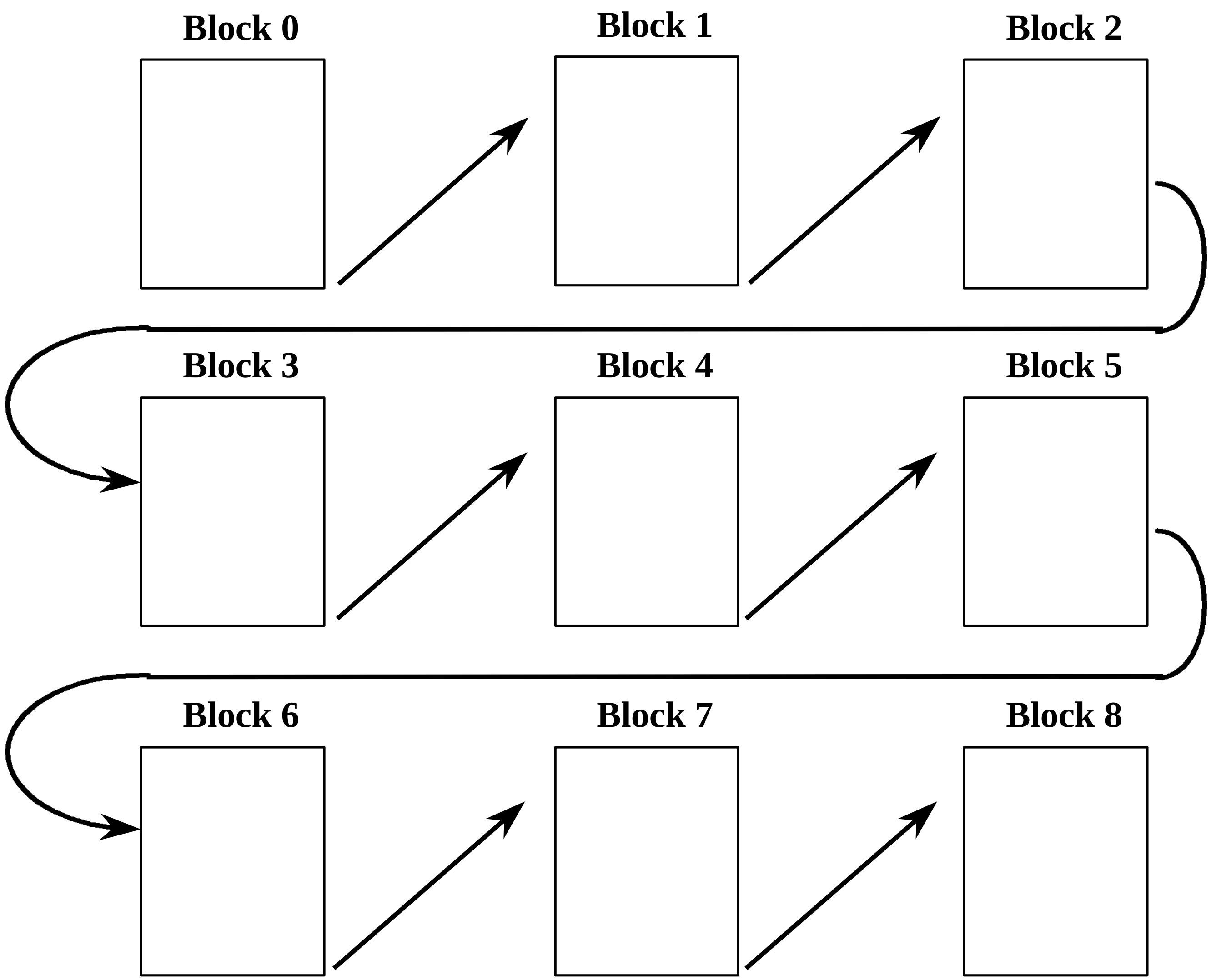
Blocked Padded Image
(0,0)
(0, C-1)
C, COLUMN
Original Pixels
Pad Pixels
B0
B1
B2
B3
B4
B5
B6
B7
B8
B9
B10
B11
(R-1, C-1)
(R-1, 0)
R, ROW

BLOCKED PADDED IMAGE WITH EMPTY BLOCKS

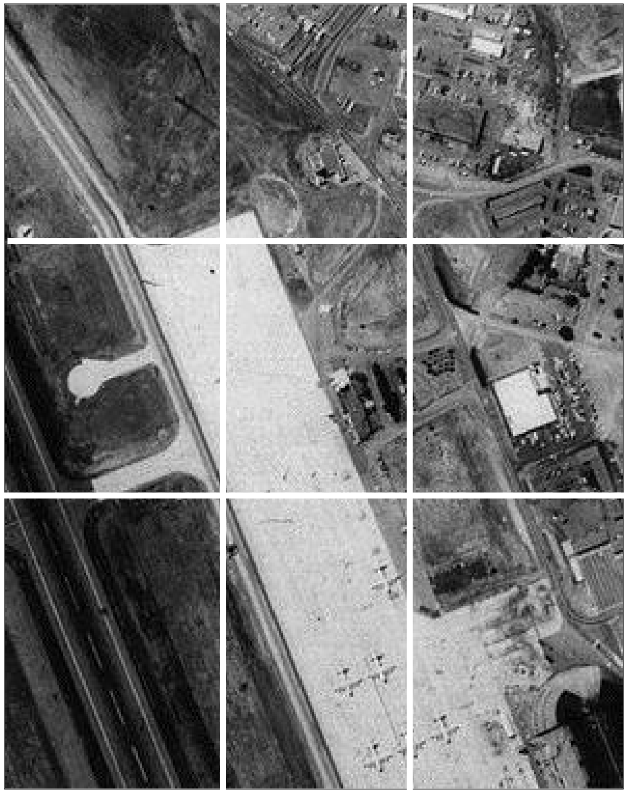


0 - 0X00000000
1 - 0X00100000
2 - 0X00200000
3 - 0XFFFFFFFF
4 - 0X00300000
5 - 0X00400000
6 - 0X00500000
7 - 0XFFFFFFFF
8 - 0X00600000
9 - 0X00700000
10 - 0X00800000
11 - 0X00900000
12 - 0XFFFFFFFF
13 - 0X00A00000
14 - 0X00B00000
15 - 0X00C00000

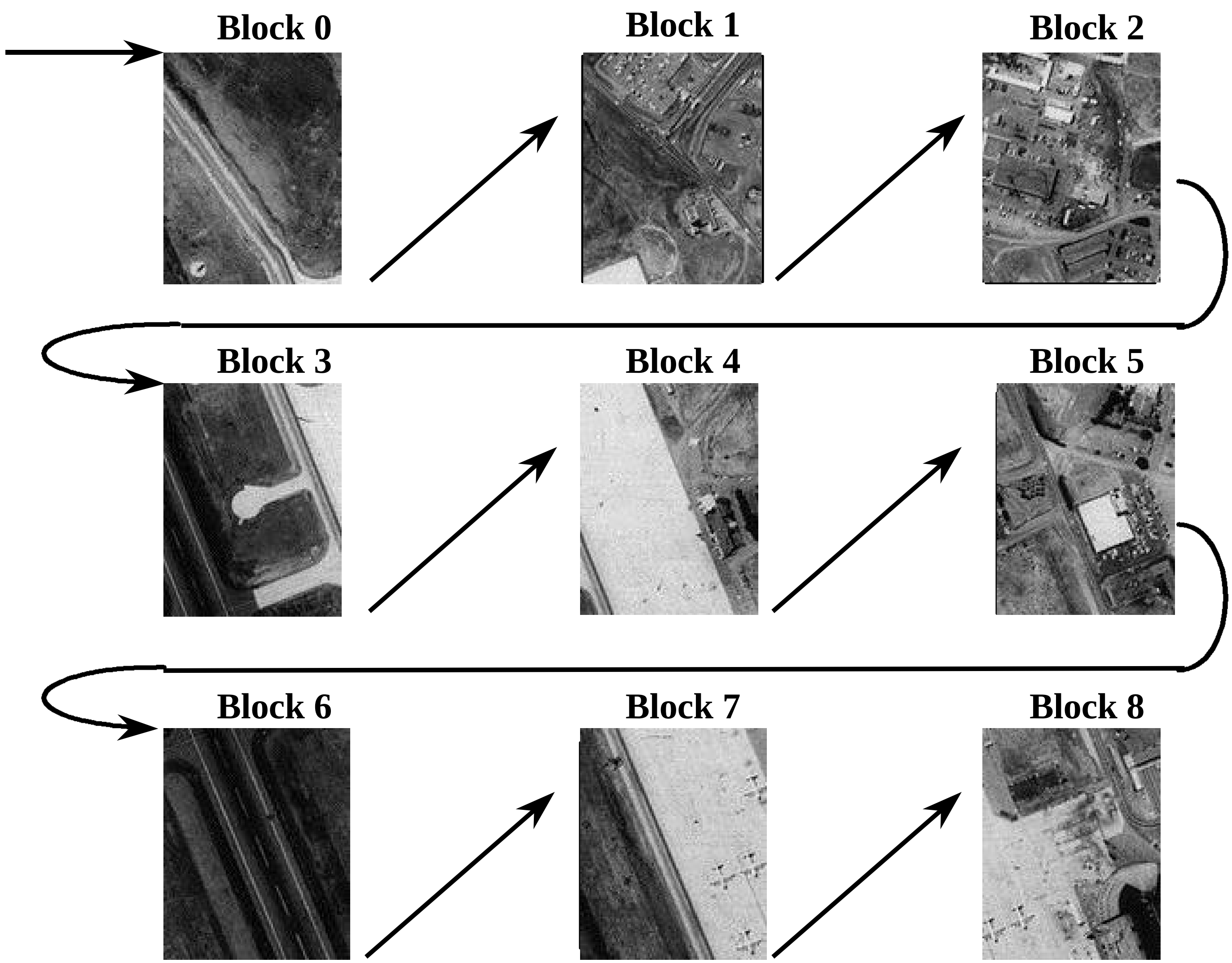
IMODE = B, BAND INTERLEAVED BY BLOCK
MONOCHROME CASE



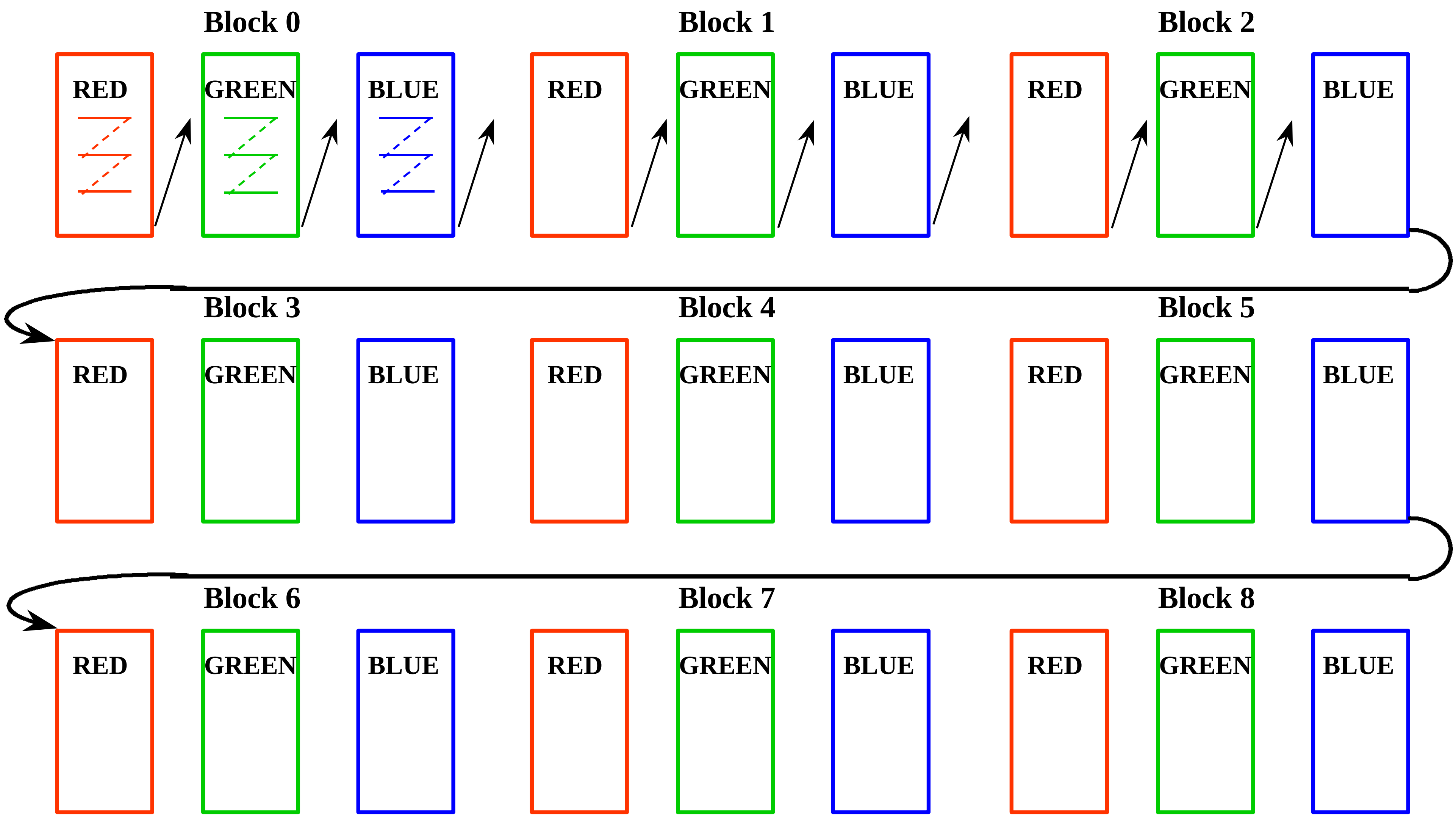
IMODE = B, BAND INTERLEAVED BY BLOCK
MONOCHROME CASE



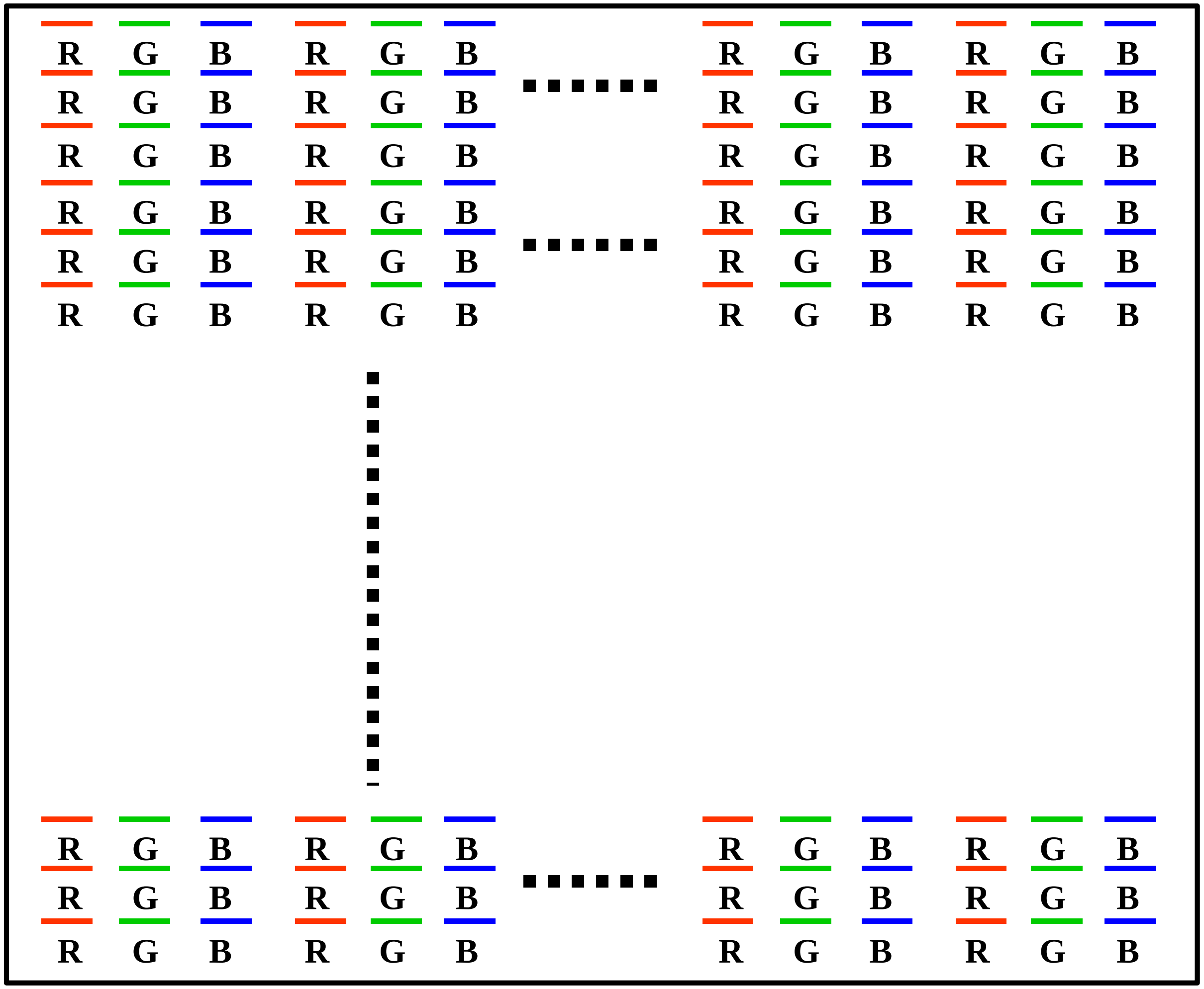
IMODE = B, BAND INTERLEAVED BY BLOCK
MONOCHROME CASE



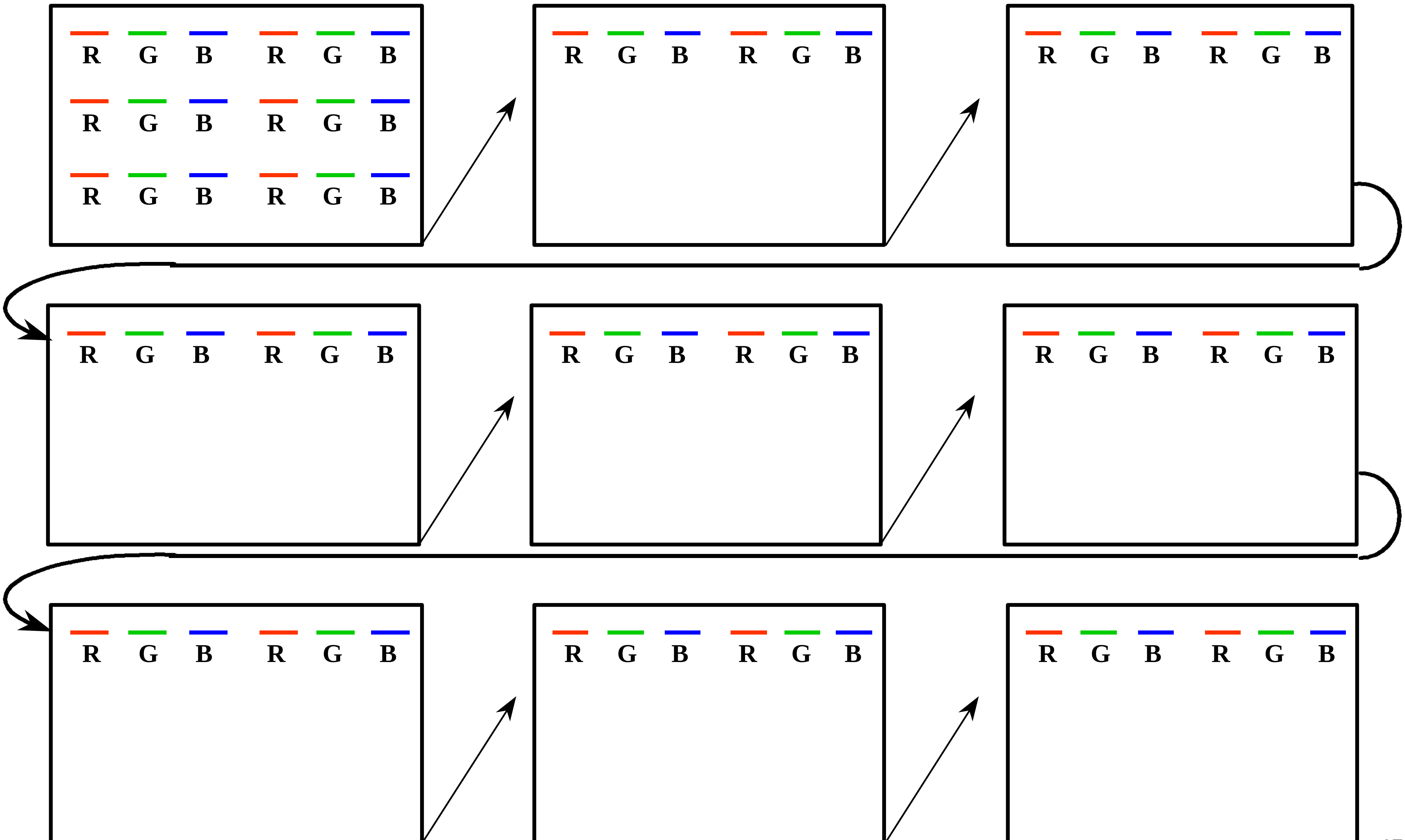
IMODE = B, BAND INTERLEAVED BY BLOCK
THREE BAND COLOR CASE



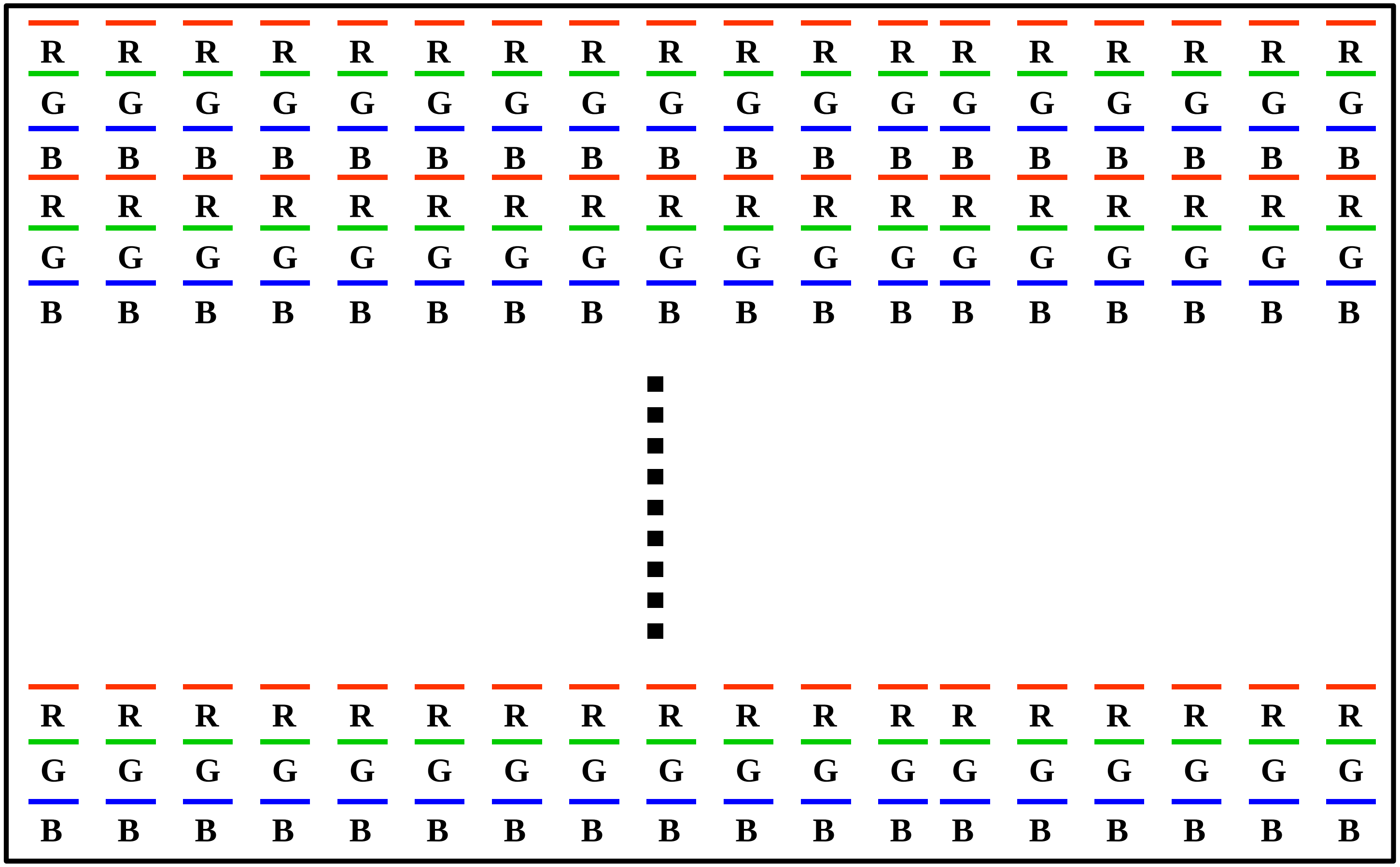
IMODE = P, BAND INTERLEAVED BY PIXEL
THREE BAND COLOR CASE



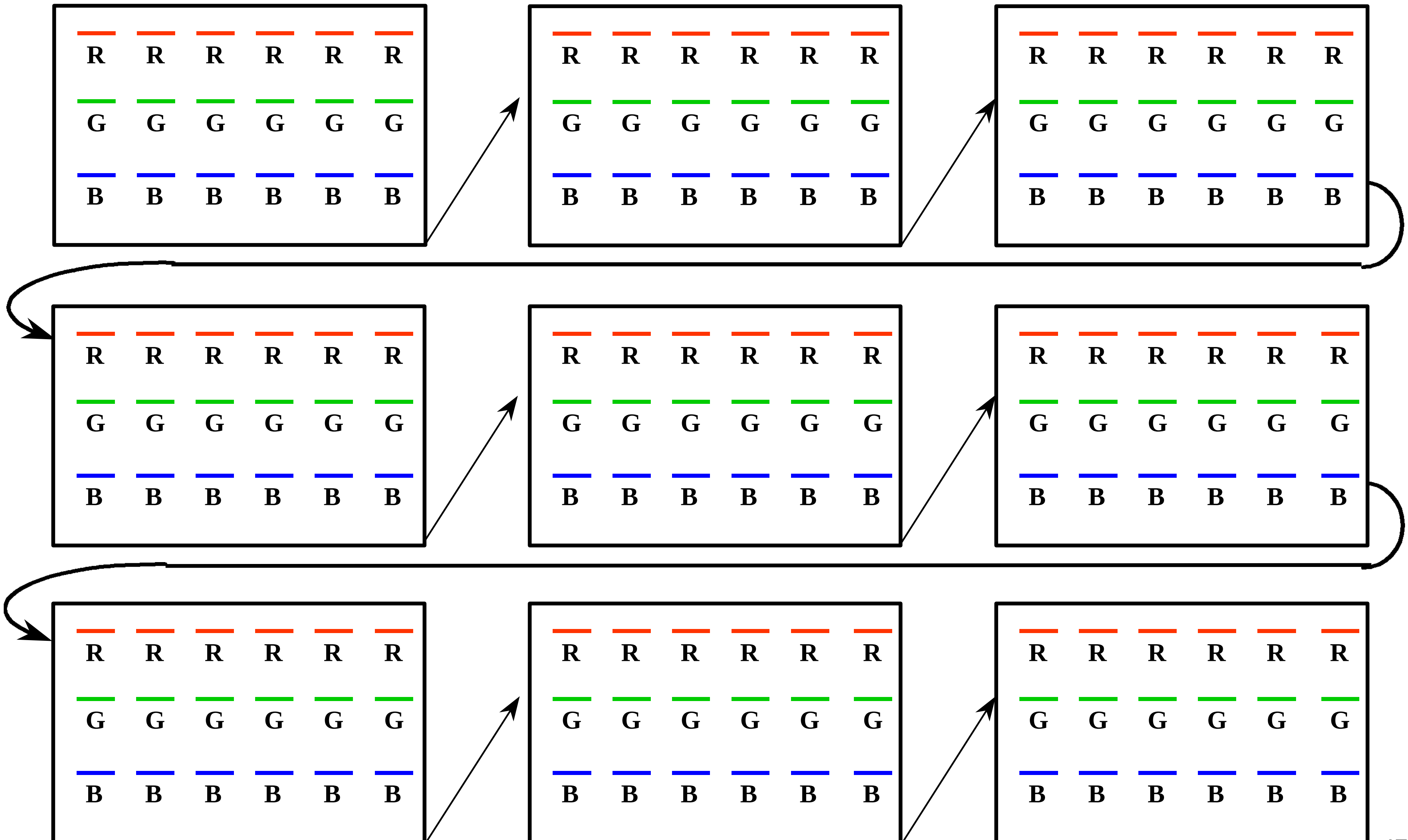
IMODE = P, BAND INTERLEAVED BY PIXEL
THREE BAND COLOR CASE



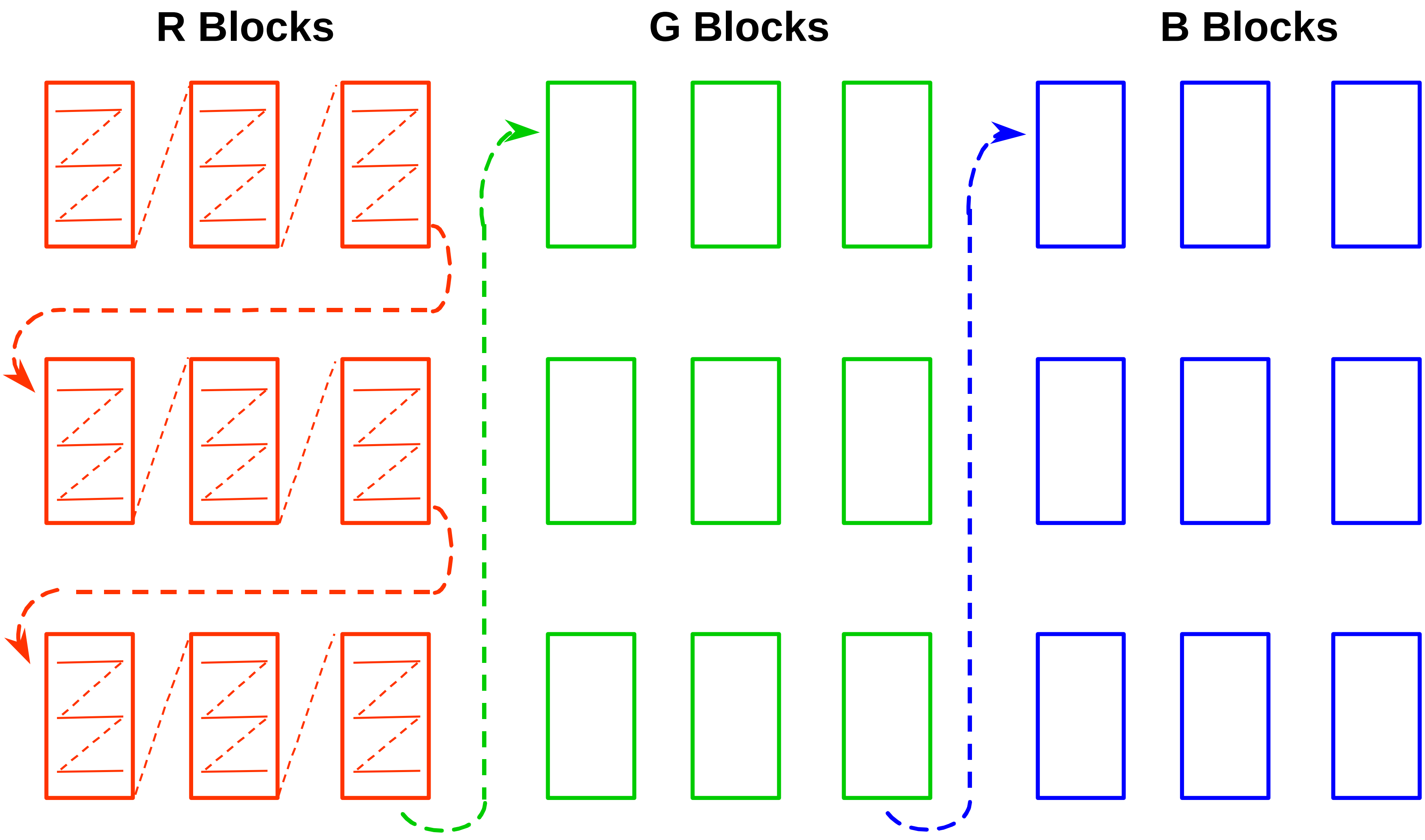
IMODE = R, BAND INTERLEAVED BY ROW
THREE BAND COLOR CASE



IMODE = R, BAND INTERLEAVED BY ROW
THREE BAND COLOR CASE

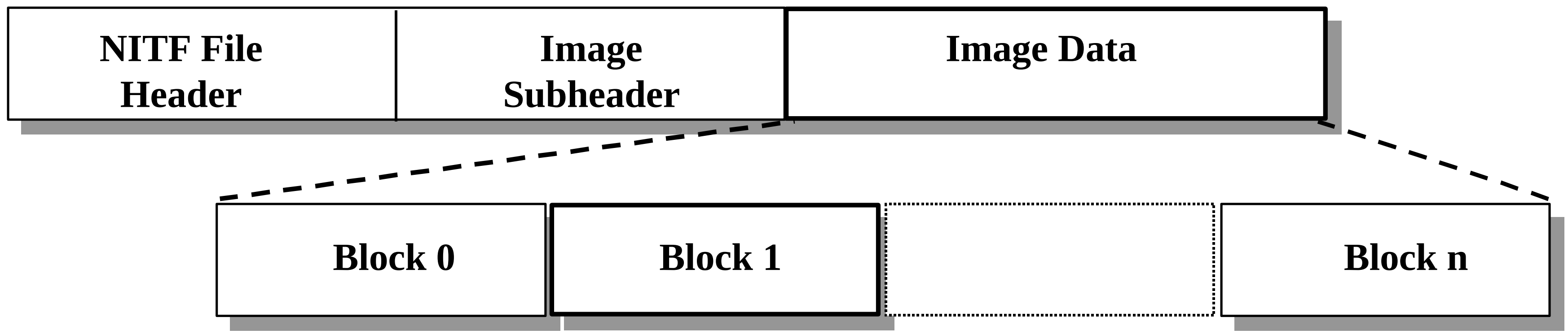


IMODE = S, BAND SEQUENTIAL
THREE BAND COLOR CASE

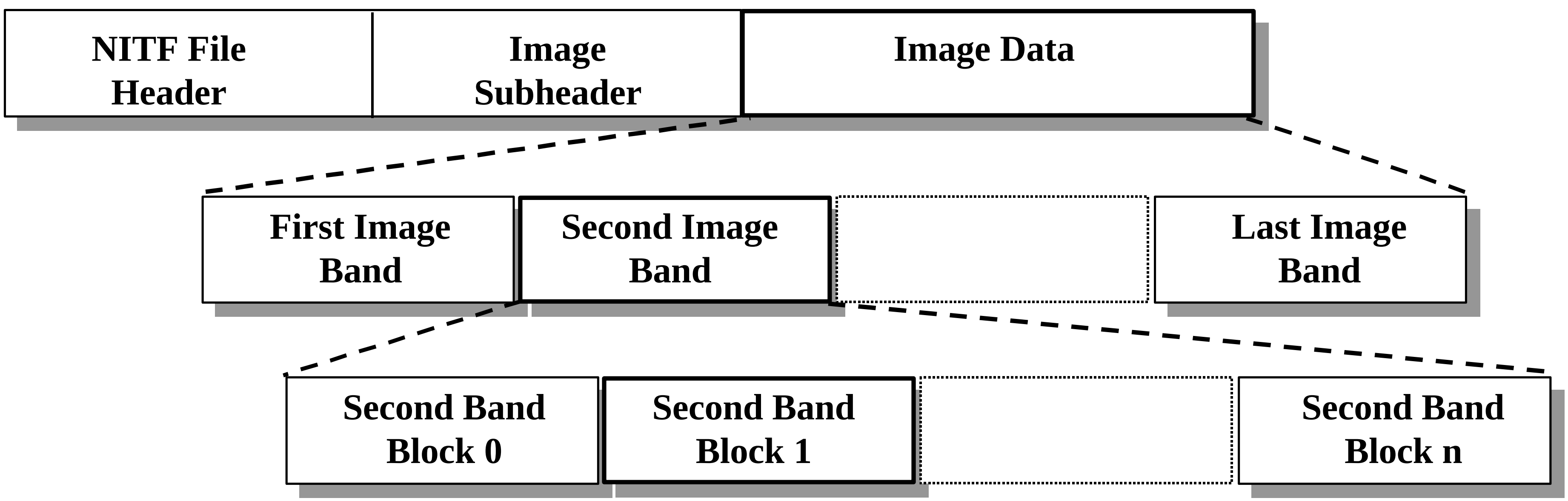


UNCOMPRESSED NITF MULTIPLE BLOCK FILE STRUCTURE

IMODE = B or P



IMODE = S



LUTS

Sample LUT

PIXEL VALUE	RED	GREEN	BLUE	MEANING
00	00000000	11111111	00000000	Green
01	00000000	00000000	11111111	Blue
10	11111111	00000000	00000000	Red
11	11111111	11111111	11111111	White

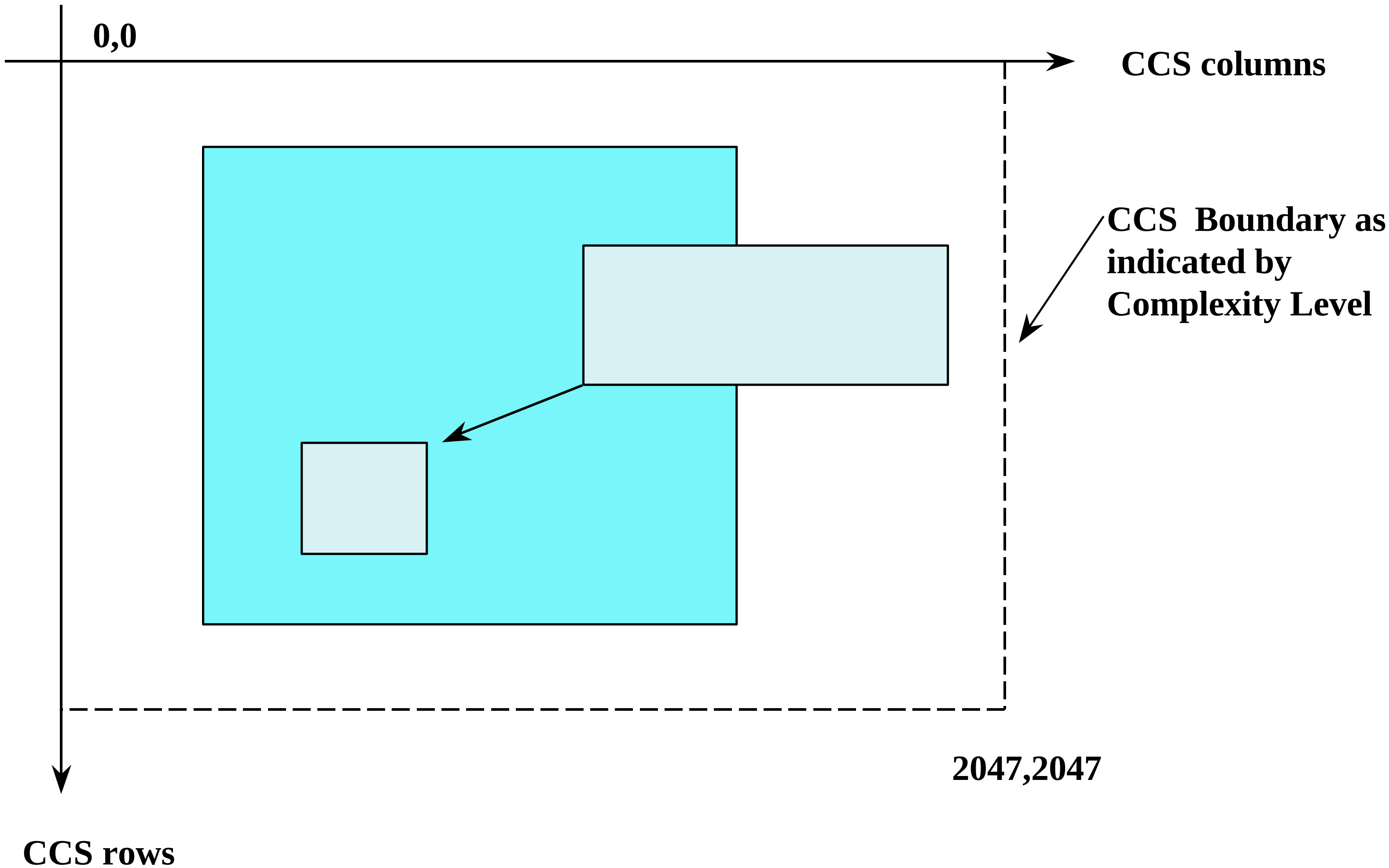
IMAGERY ANNOTATIONS

- **COMPUTER GRAPHICS METAFILE**
- **BIT-MAPPED RASTER (Legacy)**
- **LABELS (Legacy)**
- **FUTURE**
  - **VOICE**
  - **MOTION IMAGERY**
  - **ATTRIBUTES**

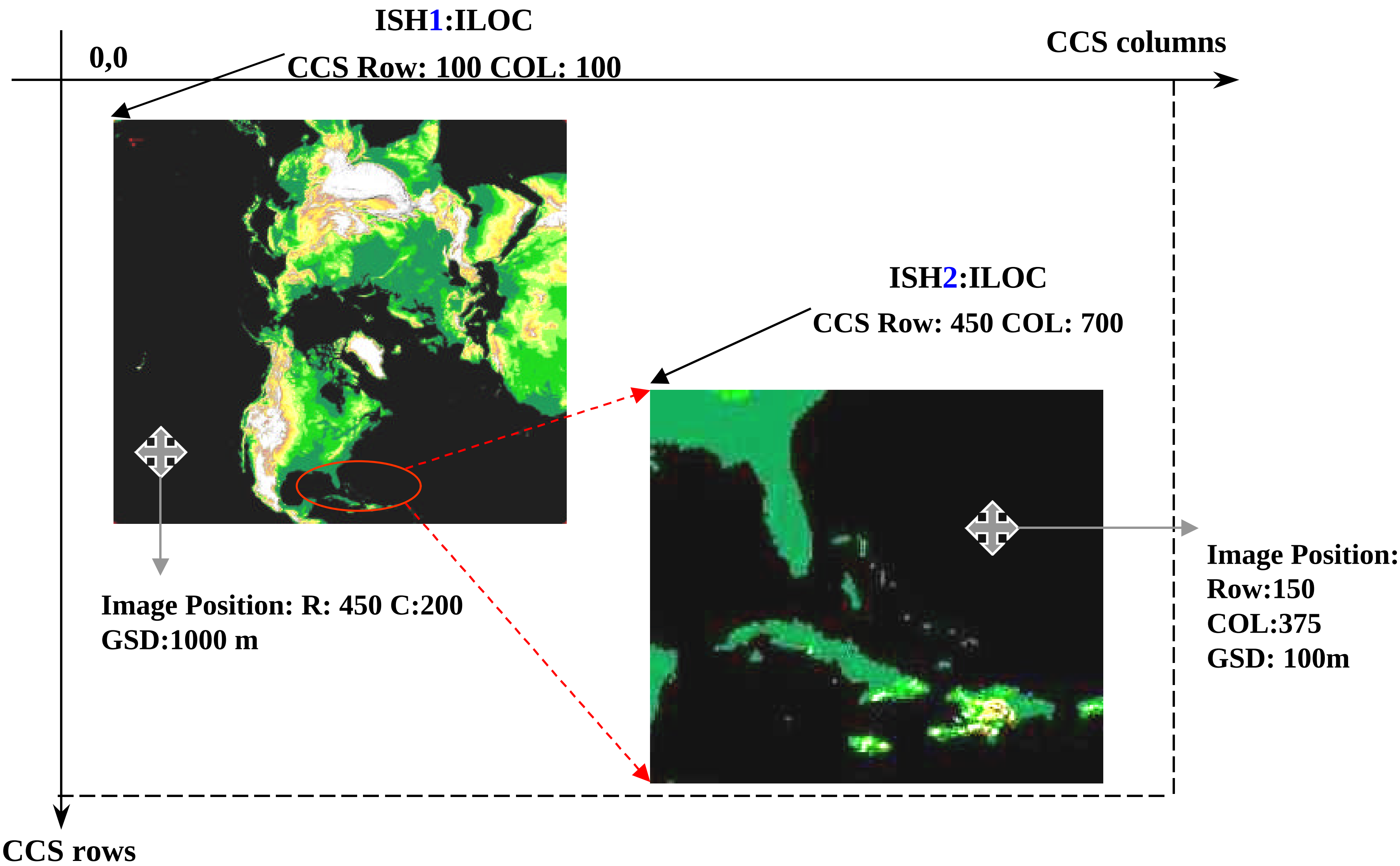
TEXTUAL ADJUNCTS

- **NITF 2.0**
  - **STA - Single byte characters, Basic Latin set**
  - **MTF - MIL-STD-6040, USMTF**
    - **For example, GRAPHREP Messages**
- **NITF 2.1 includes above and adds:**
  - **UT1**
    - **Single byte characters**
    - **Basic Latin character set**
    - **Latin Supplement 1 (LS-1) character set**
  - **U8S**
    - **UTF-8 character encoding**
    - **Basic Latin and LS-1 character set**

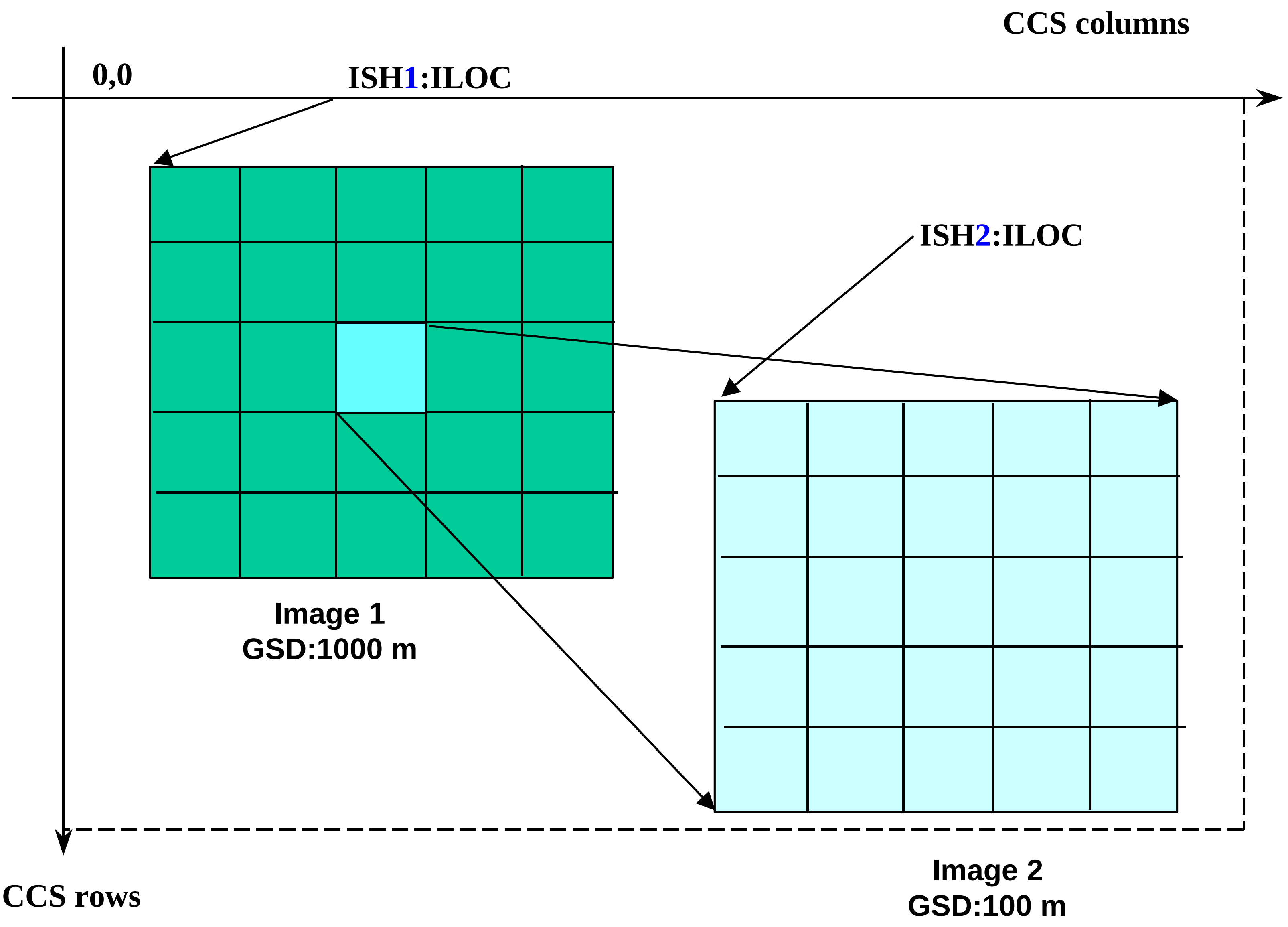
COMMON COORDINATE SYSTEM



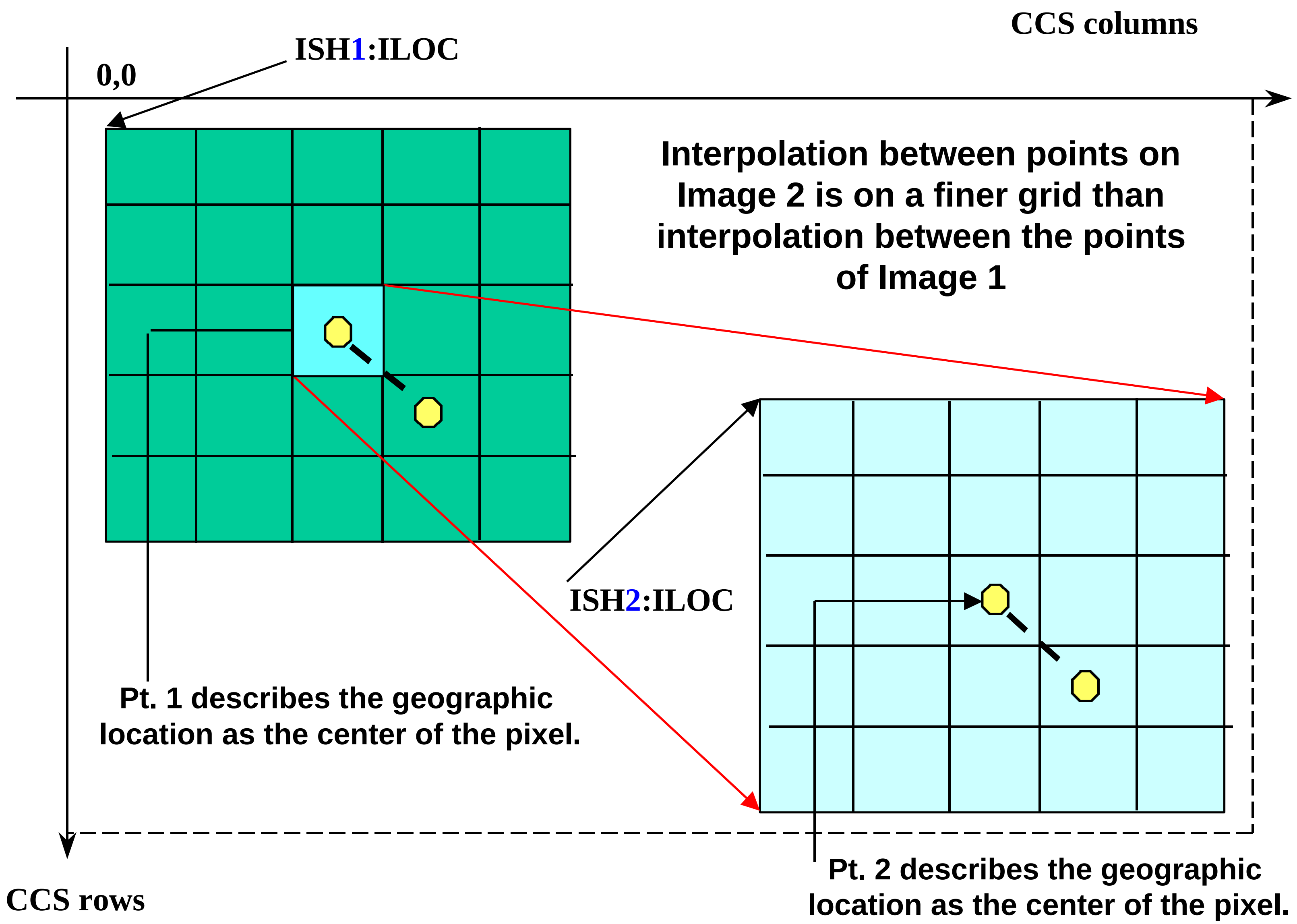
COMMON COORDINATE SYSTEM



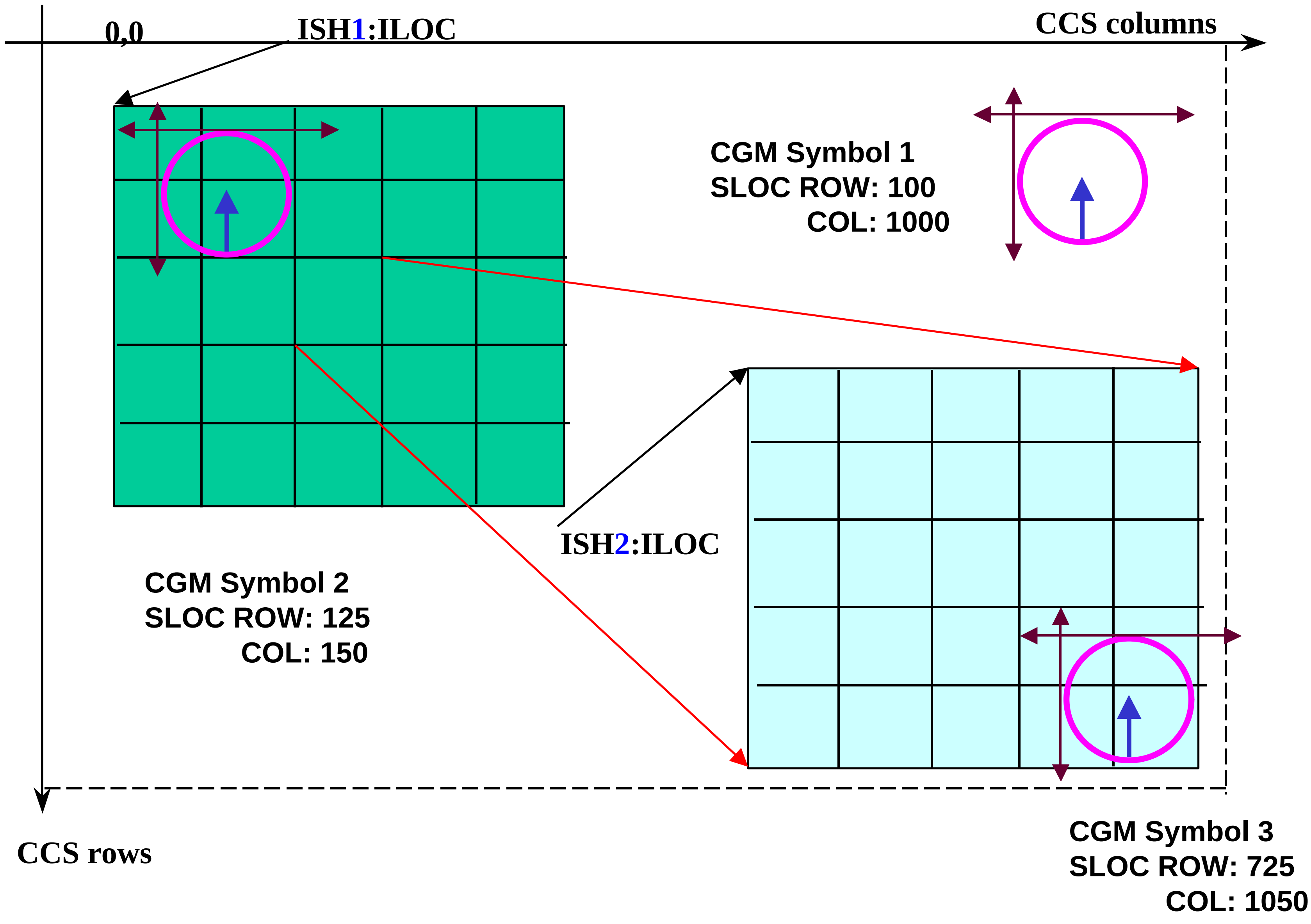
COMMON COORDINATE SYSTEM



COMMON COORDINATE SYSTEM

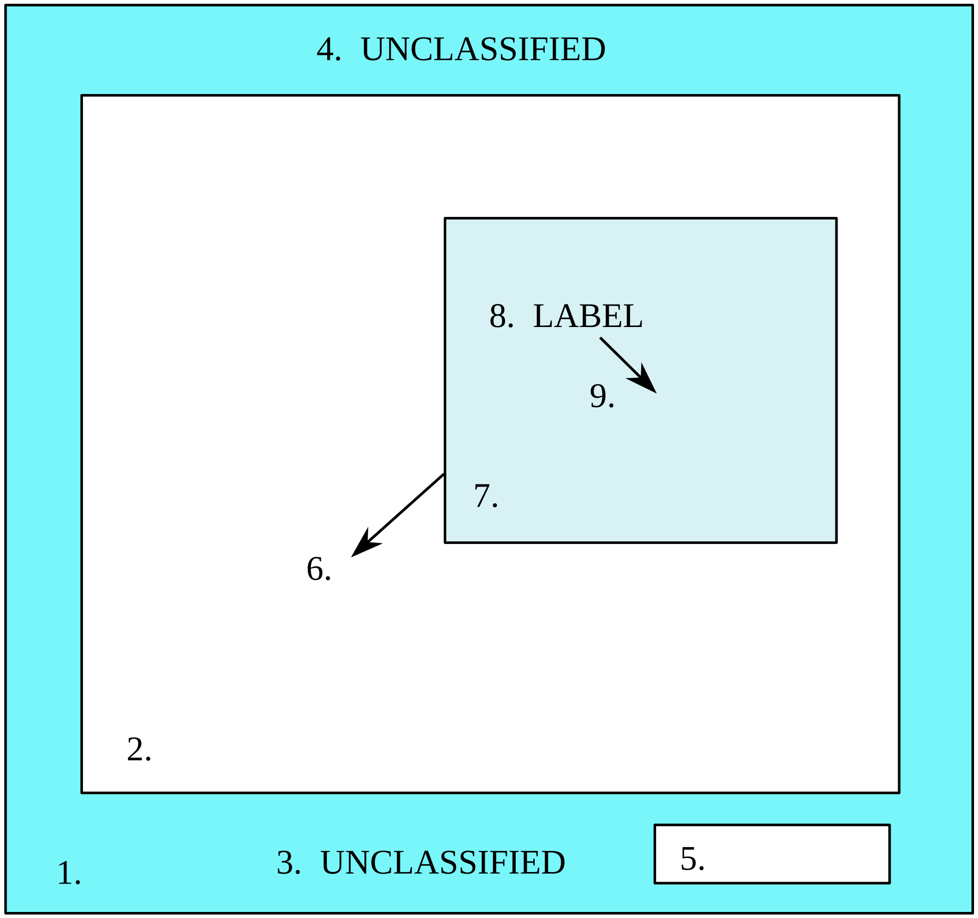


COMMON COORDINATE SYSTEM



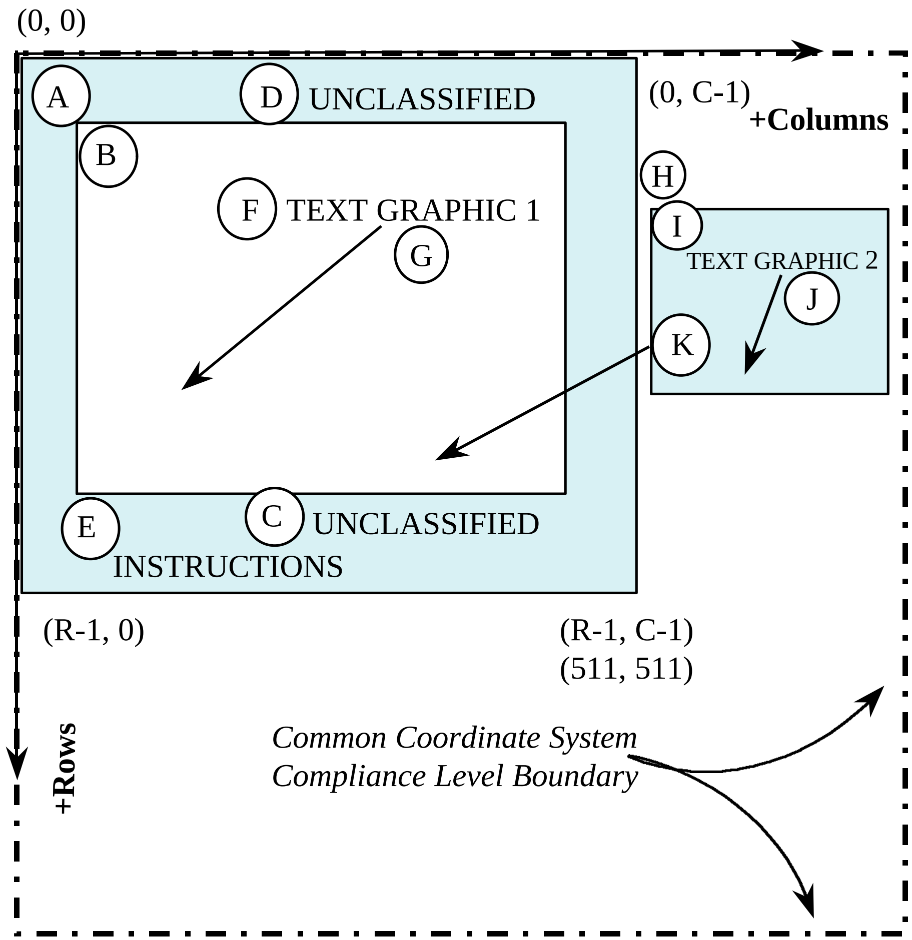
DISPLAY LEVEL ILLUSTRATION

1. Border (an opaque box)	DL = 001
2. Exploited Image	DL = 002
3. Classification Marking 1	DL = 999
4. Classification Marking 2	DL = 998
5. Handling Instructions	DL = 997
6. Arrow 1 (image inset)	DL = 003
7. Image Inset	DL = 004
8. Label	DL = 005
9. Arrow 2 (label)	DL= 006



COMMON COORDINATE WITH DISPLAY & ATTACHMENT LEVELS

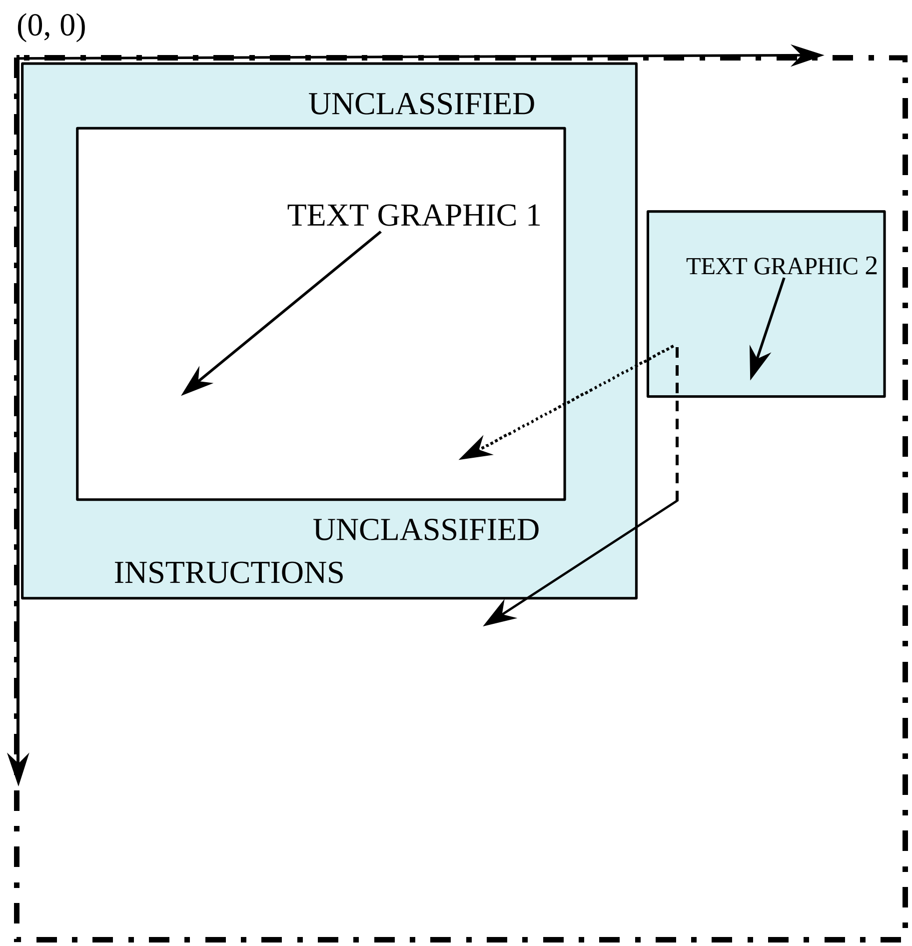
			Location Offsets	
	DL	**AL**	**Row**	**Column**
A. Border (opaque box)	*001*	*000*	*00000*	*00000*
B. Exploited Image	002	001	00025	00025
C. Classified Marking 1	999	000	00420	00256
D. Classified Marking 2	998	000	00015	00256
E. Handling Instructions	997	000	00440	00040
F. Text Graphic 1	003	002	00060	00240
G. Arrow 1	004	003	00010	00040
H. Image Inset	005	002	00100	00525
I. Text Graphic 2	006	005	00020	00050
J. Arrow 2	007	006	00010	00050
K. Arrow 3	008	005	00070	-0005



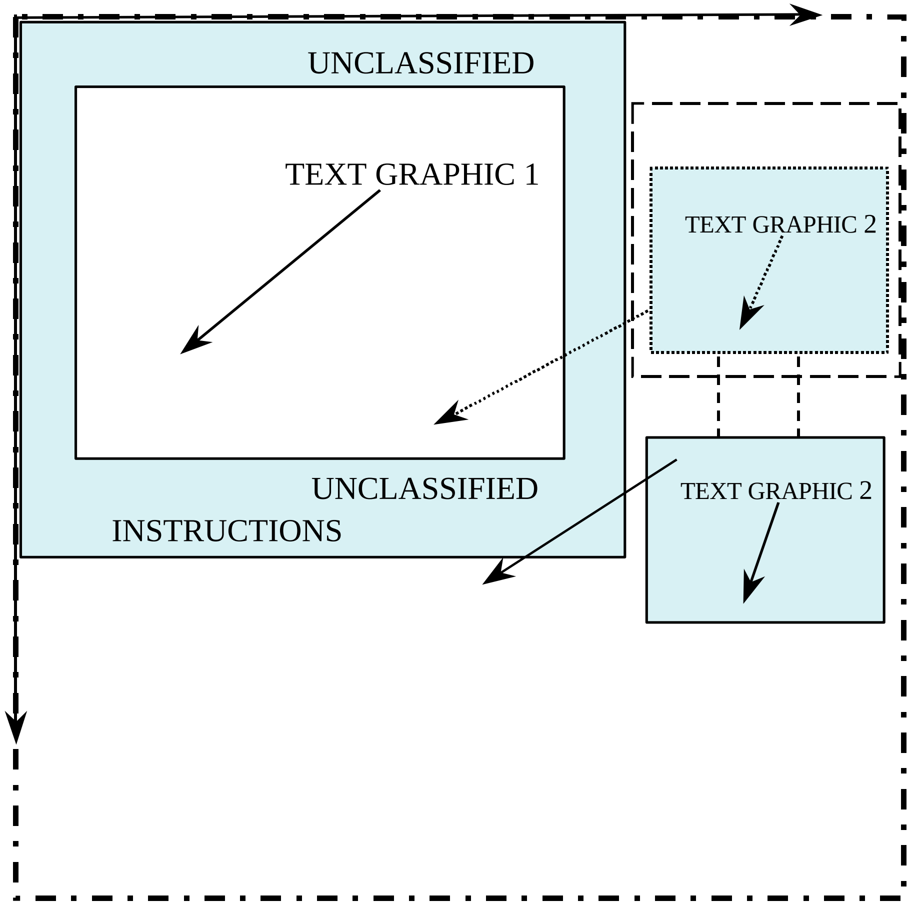
PARENT CHILD RELATIONSH IP

- Screen Display DL 000, AL 000
  - A. Border 001, 000
    - B. Exploited Image 002, 001
      - F. Text Graphic 1 003, 002
        - G. Arrow 1 004, 003
      - H. Image Inset 005, 002
        - I. Text Graphic 1 006, 005
          - J. Arrow 2 007, 006
        - K. Arrow 3 008, 005
  - C. Class Mark Bot 999, 000
  - D. Class Mark Top 998, 000
  - E. Handle Inst. 997, 000

DISPLAY & ATTACHMENT LEVEL PARENT MOVE



DISPLAY & ATTACHMENT LEVEL PARENT CHILD MOVE



DATA EXTENSIONS

- **Flexibility**
- **Extensibility**
- **Types of Extensions: TRE (tag), DES, RES**
- **Example:**

TRE: Sensors collecting imagery also collect and report auxiliary data that uniquely identifies the imagery, defines the collection geometry and contains other information to aid exploitation.

DES: Desire to add a new data type for inclusion within an NITF file, perhaps in conjunction with other data types, e.g. a DES for motion imagery.

FORMAT DETAILS

- Complexity Levels
- File Header
- Security Markings
- Image Subheader
  - IREP
  - ICAT
  - PVTYPE
- Image Geographic Location

COMPLEXITY LEVELS

- **This concept allows NITFS to be implemented on a wide range of hardware platforms with various levels of resources**

NITF 2.1 CLEVEL	NITF 2.0 CLEVEL
07	N/A
06	06
05	05, 04
03	03, 02, 01

- **Implementations of the NITFS are tested according to their ability to pack and/or unpack various CLEVELS of NITFS formatted files**

- **Files shall be marked at the lowest CLEVEL for which they qualify**

COMPLEXITY LEVELS

NITF FILE Features	Complexity Level			
	03	05	06	07
Common Coordinate System Extent (Pixels)	(00000000, 00000000) to (00002047, 00002047)	(00000000, 00000000) to (00008191, 00008191)	(00000000, 00000000) to (00065535, 00065535)	(00000000, 00000000) to (99999999, 99999999)
Maximum File Size	50 Mbyte - 1 byte (52,428,799 bytes)	1 Gbyte - 1byte (1,073,741,823 bytes)	2 Gbyte -1byte (2,147,483,647 bytes)	10 Gbyte - 1 byte (10,737,418,239 bytes)
Image Size (Image(s) placed within CCS extent)	00000002 to 00002048 Rows X 00000002 to 00002048 Columns (R and C $\leq$ 2048)	00000002 to 00008192 Rows X 00000002 to 00008192 Columns (R or C >2048)	00000002 to 00065536 Rows X 00000002 to 00065536 Columns (R or C > 8192)	00000002 to 99999999 Rows X 00000002 to 99999999 Columns (R or C > 65536)
Image Blocking (Rectangular Blocks allowed)	Single and Multiple Blocks 0001 to 2048 Rows X 0001 to 2048 Cols	Single and Multiple Blocks 0001 to 8192 Rows X 0001 to 8192 Cols	Multiple blocking is mandatory for images that exceed 8192 pixels per Row or Column 0001 to 8192 Rows X 0001 to 8192 Columns	
Multispectral (MULTI) No Compression	2 to 9 Bands 8,16, 32, 64-Bits per pixel per band With and without LUT in each band IC=NC, NM IMODE = B, P R, S	2 to 255 Bands, 8, 16, 32, 64-Bits per pixel per band With and without LUTs in each band IC=NC, NM IMODE = B, P R, S		2 to 999 Bands 8,16, 32, 64-Bits per pixel per band With and without LUTs in each band IC=NC, NM IMODE = B, P R, S

COMPLEXITY LEVELS (CONT'D)

NITF FILE Features	Complexity Level			
	03	05	06	07
Multispectral (MULTI) Individual Band JPEG Compression	2 to 9 Bands 8, 12-Bits per Pixel per Band No LUT IMODE = B, S IC = C3, M3	2 to 255 Bands, 8, 12 Bits per Pixel per Band No LUT IMODE = B, S IC = C3, M3		2 to 999 Bands 8, 12-bits per Pixel per Band No LUT IMODE = B, S IC = C3, M3
Multispectral (MULTI) Multi-Component Compression	2 to 9 Bands 8, 12-Bits per Pixel per Band No LUT IMODE = B, P, S IC = C6, M6 (This feature is optional for implementation)	2 to 256 Bands, 8, 12-Bits per Pixel per Band With and without LUT in each Band IMODE = B, P, S IC = C6, M6 (This feature is optional for implementation)		2 to 999 Bands 8, 12-Bits per Pixel per Band No LUT IMODE = B, P, S IC = C6, M6 (This feature is optional for implementation)
Matrix Data (NODISPLY)	1 to 9 Bands 8, 16, 32, 64-Bits per pixel per band No LUT in any band IMODE = B.P, R, S (This feature is optional for implementation)	1 to 255 Bands, 8, 16, 32, 64-Bits per pixel per band No LUT in any band IMODE = B, P, R, S (This feature is optional for implementation)		1 to 999 Bands 8, 16, 32, 64-Bits per pixel per band No LUT in any band IMODE = B, P, R, S (This feature is optional for implementation)

COMPLEXITY LEVELS (CONT'D)

NITF FILE Features	Complexity Level			
	03	05	06	07
Number of Image Segments Per File	0 - 20	0 - 100		
Aggregate Size of Graphic Segments Per File	1 Mbyte maximum	2 Mbyte maximum		

NITF 2.1 FILE HEADER

FIELD	SIZE	VALUE RANGE	TYPE
FHDR	4	NITF	R
FVER	5	02.10	R
CLEVEL	2	03, 05, 06, 07	R
STYPE	4	BF01	R
OSTAID	10	(Originating Station Identifier)	R
FDT	14	CCYYMMDDhhmmss	R
FTITLE	80	(Default is spaces)	R
FSCLAS	1	T, S, C, R, or U	R
SECURITY GROUP	166	(Detail shown in another table)	R
FSCOP	5	00000-99999, (Default is 00000)	R
FSCPYS	5	00000-99999, (Default is 00000)	R

NITF File Header	Image Segments	Graphic Segments	Text Segments	Data Extension Segments	Reserved Extension Segments

NITF 2.1 FILE HEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
ENCRYP	1	0 = Not Encrypted	R
FBKGC	3	Unsigned binary integer for RGB color (0x00-0xFF, 0x00-0xFF, 0x00-0xFF)	R
ONAME	24	(Default is spaces)	R
OPHONE	18	(Default is spaces)	R
FL	12	000000000388-999999999998, 999999999999	R
HL	6	000388-999999	R
NUMI	3	(Default is BCS zeros) 000-999	R
LISHn	6	000439-999998, 999999	C
LIn	10	0000000001-9999999998, 9999999999	C

NITF 2.1 FILE HEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
NUMS	3	(Default is BCS zeros) 000-999	R
LSSHn	4	0258-9998, 9999	C
LSn	6	000001-999998, 999999	C
NUMX	3	000	R
NUMT	3	(Default is BCS zeros) 000-999	R
LTSHn	4	0282-9998, 9999	C
LTn	5	00001-99998, 99999	C
NUMDES	3	(Default is BCS zeros) 000-999	R
LDSHn	4	0200-9998, 9999	C
LDn	9	000000001-999999998, 999999999	C

NITF 2.1 FILE HEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
NUMRES	3	(Default is BCS zeros) 000-999	R
LRESHn	4	0200-9999	C
LREn	7	0000001-9999999	C
UDHDL	5	(Default is BCS zeros) 00000 or 00003-99999	R
UDHOFL	3	(Default is BCS zeros) 000-999	C
UDHD	*[1]	Tagged Record Extensions	C
XHDL	5	(Default is BCS zeros) 00000 or 00003-99999	R
XHDLOFL	3	(Default is BCS zeros) 000-999	C
XHD	**[1]	Tagged Record Extensions	C

*1 A value specified in the UDHDL field minus 3 (in bytes)

**1 A value specified in the XHDL field minus 3 (in bytes)

NUMI EXAMPLE

- NUMI = 3
  - LISH001 = 000XXX
  - LI001 = 00000YYYYY
  - LISH002 = 000XXX
  - LI002 = 0000000YYY
  - LISH003 = 000XXX
  - LI003 = 0000000YYY

Image Segment #1		Image Segment #2		Image Segment #3	
000XXX Image #1 Subheader	00000YYYYY Image #1 Image Data	000XXX Image #2 Subheader	0000000YYY Image #2 Image Data`	000XXX Image #3 Subheader	0000000YYY Image #3 Image Data

SECURITY

- Main Header contains overall security classification of file.
- Each data segment has individual security classification.
- Security group within the Main Header and each data segment contains:

NITF 2.0	NITF 2.1
– Classification	– Classification
– Codewords	– Codewords
– Control and Handling	– Control and Handling
– Releasing Instructions	– Releasing Instructions
– Classification Authority	– Declassification Type
– Security Control Number	– Declassification Date
– Security Downgrading Instructions	– Declassification Exemption
	– Downgrade
	– Downgrade Date
	– Classification Text
	– Classification Authority Type
	– Classification Authority
	– Classification Reason
	– Security Source Date
	– Security Control Number

SECURITY (CONT'D)

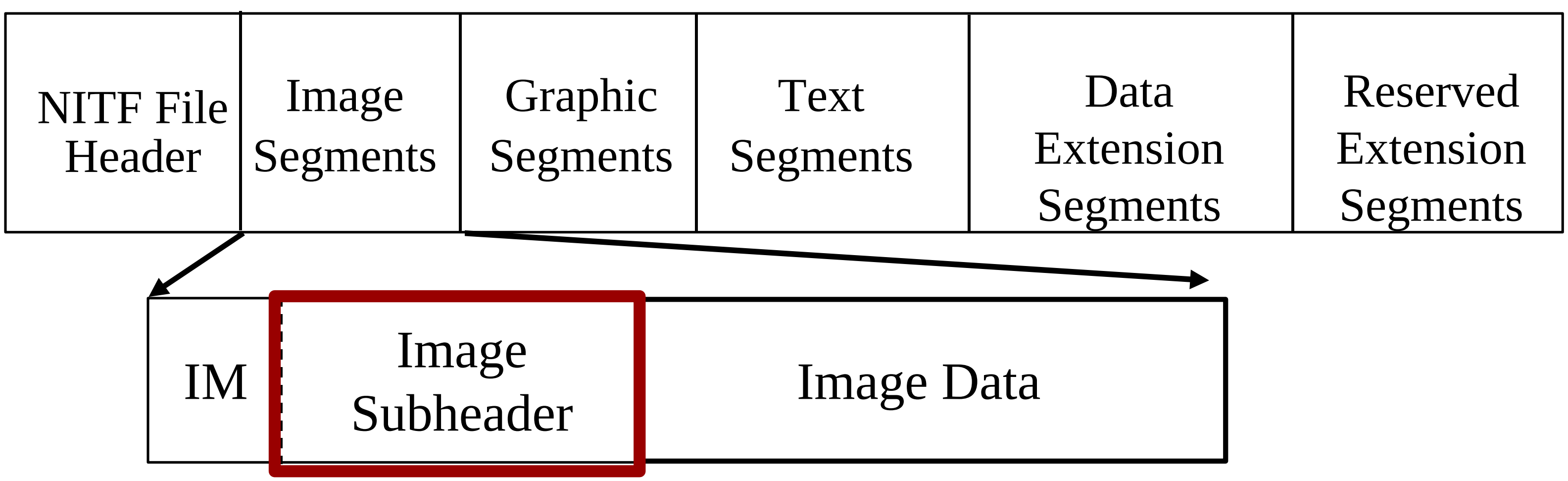
System Security Group -166 bytes		
XSCLSY	2 bytes	Security Classification System
XSCODE	11 bytes	Codewords
XSCTLH	2 bytes	Control and Handling
XSREL	20 bytes	Releasing Instructions
XSDCTP	2 bytes	Declassification Type, DD, DE, GD, GE, O, or X
XSDCDT	8 bytes	Declassification Date, CCYYMMDD
XSDCXM	4 bytes	Declassification Exemption, X1-X8 and X251-X259
XSDG	1 byte	Downgrade, S, C, or R

SECURITY (CONT'D)

System Security Group -166 bytes (Cont'd)		
XSDGDT	8 bytes	Downgrade Date, CCYYMMDD
XSCLTX	43 bytes	Classification Text
XSCATP	1 byte	Classification Authority Type, O or D
XSCAUT	40 bytes	Classification Authority
XSCRSN	1 byte	Classification Reason, A-G
XSSRDT	8 bytes	Security Source Date, CCYYMMDD
XSCTLN	15 bytes	Security Control Number

2.1 IMAGE SUBHEADER

FIELD	SIZE	VALUE RANGE	TYPE
IM	2	IM	R
IID1	10	User defined	R
IDATIM	14	CCYYMMDDhhmmss	R
TGTID	17	BBBBBBBBBBOOOOOCC (Default is BCS spaces)	R
IID2	80	(Default is BCS spaces)	R
ISCLAS	1	T, S, C, R, or U	R
Security Group	166	(Detail shown in another table)	R



2.1 IMAGE SUBHEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
ENCRYP	1	0=Not Encrypted	R
ISORCE	42	(Default is BCS spaces)	R
NROWS	8	00000002-99999999	R
NCOLS	8	00000002-99999999	R
PVTYPE	3	INT, B, SI, R, C	R
IREP	8	MONO, RGB, RGB/LUT, MULTI, NODISPLY, NVECTOR, POLAR, VPH, YCbCr601	R
ICAT	8	VIS, SL, TI, FL, RD, EO, OP, HR, HS, CP, BP, SAR, SARIQ, IR MAP, MS, FP, MRI, XRAY, CAT, VD, PAT, LEG, DTEM, MATR, LOCG, BARO, CURRENT, DEPTH, WIND (Default is VIS)	R

2.1 IMAGE SUBHEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
ABPP	2	01-96	R
PJUST	1	L or R (Default is R)	R
ICORDS	1	U, G, N, S, D, or space (Default is BCS spaces)	R
IGEOLO	60	±dd.ddd±ddd.ddd (four times) or ddmmssXdddmmssY (four times) or zzBJKeeeeennnnn (four times) or zzeeeeeennnnnnn (four times)	C
NICOM	1	0-9	R
ICOMn	80	User defined	C

2.1 IMAGE SUBHEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
IC	2	NC, NM, C1, C3, C4, C5, C6, C7, C8 M1, M3, M4, M5, M6, M7, M8	R
COMRAT	4	(Depends on the value of the IC field)	C
NBANDS	1	0-9	R
XBANDS	5	00010-99999	C
IREPBANDn	2	Standard values are LU, R, G, B, M, Y, Cb, Cr Additional values are allowed through the registration process	R
ISUBCATn	6	I, Q, M, P, SPEED, DIRECT, User defined (Default is spaces)	R
IFCn	1	N	R
IMFLTn	3	Fill with BCS spaces	R

2.1 IMAGE SUBHEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
NLUTSn	1	0-4	R
NELUTn	5	00001-65536	C
LUTDnm	★[3]	Unsigned binary integer, LUT Values	C
ISYNC	1	0 = No Sync Code	R
IMODE	1	B, P, R, S	R
NBPR	4	0001-9999	R
NBPC	4	0001-9999	R
NPPBH	4	0001-8192	R
NPPBV	4	0001-8192	R
NBPP	2	01-96	R

2.1 IMAGE SUBHEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
IDLVL	3	001-999	R
IALVL	3	000-998 (Default is BCS zeros)	R
ILOC	10	00000 to 99999 or -0001 to -9999	R
IMAG	4	decimal value, /2 followed by 2 spaces, /4 followed by 2 spaces, /8 followed by 2 spaces, /16 followed by a space, /32 followed by a space, /64 followed by a space, or /128 (Default is 1.0 followed by a space)	R

2.1 IMAGE SUBHEADER (CONT'D)

FIELD	SIZE	VALUE RANGE	TYPE
UDIDL	5	00000 or 00003-99999	R
UDOFL	3	000-999	C
UDID	*[3]	Tagged Record Extensions	C
IXSHDL	5	00000 or 00003-99999	R
IXSOFL	3	000-999	C
IXSHD	**[3]	Tagged Record Extensions	C

H3 One Byte for each entry
*3 As specified in UDIDL
**3 As specified in IXSHDL

PIXEL VALUE TYPE

- PVTYPE - Indicates the computer representation used for the value of each pixel or:
  - INT - Integer
  - SI - Signed Integer
  - B - Binary
  - C - Complex
  - R - Real

IMAGE REPRESENTATION

IREP - Indicates the processing required to display the image.

- MONO
- RGB
- YCbCr601
- RGB/LUT
- MULTI
- NODISPLY

IMAGE REPRESENTATION

IREP	PVTYPE	ICAT
MONO	B or INT	VIS, OP, SL, TI, FL, RD, EO, HR, BP, FP, VD, SAR, IR
RGB/LUT	INT	VIS, OP, MAP, LEG, CP, PAT, EO
RGB	INT	VIS, OP, MAP, LEG, CP, PAT, EO
YCbCr	INT	VIS, OP, MAP, LEG, CP, PAT, EO
MULTI	INT, SI, R	HS, MS
NODISPLY	INT, SI, R, C	DTEM, LOCG, MATRIX, SARIQ, BARO, CURRENT, DEPTH, WIND

IMAGE CATEGORY

ICAT - Indicates the specific category of image, raster or grid data, or the use or nature of the collector.

- VIS - Visible Imagery
- SL - Side Looking Radar
- TI - Thermal Infrared
- FL - Forward Looking
- MS - Multispectral
- HS - Hyperspectral
- CP - Color Frame Photography
- VD - Video
- RD - Radar
- EO - Electro-Optical
- OP - Optical
- HR - High Resolution Radar
- BP - Black and White Frame Photography
- SAR - Synthetic Aperture Radar

IMAGE CATEGORY (CONT'D)

- SARIQ - Synthetic Aperture Radar radio hologram
- PAT - Colour patch
- DTEM - Elevation model data
- MATR - General matrix data
- BARO - Barometric
- CURRENT - Current
- DEPTH - Depth
- WIND - Wind
- LOCG - Location grids

IGEOLO/ICORDS

COMPRESSION

- OBJECTIVES OF COMPRESSION
- BI-LEVEL
- JPEG
- VQ
- ARIDPCM
- JPEG 2000
- FUTURE

OBJECTIVES OF IMAGE COMPRESSION

- Image compression is the art/science of finding efficient representations for digital images in order to:
  - Reduce the memory required for their storage,
  - Reduce the effective data access time when reading from storage devices,
  - Reduce the bandwidth and/or the time required for their transfer across communication channels.
- The goal is to achieve the desired bit rate without compromising the image quality required for a given application.

WHY CAN IMAGES BE COMPRESSED?

- **Redundancy**
  - Redundancy relates to the statistical and information theoretic properties of an image.
  - Redundancy can be found in various forms:
    - Spatial (between neighboring pixels),
    - Spectral (between color planes), and
    - Temporal (between neighboring sequential frames in a sequence).
- **Irrelevancy**
  - Irrelevancy relates to perception and the limitations of the interpreting system, e.g. the human visual system (HVS)
  - The HVS exhibits sensitivity variations as a function of:
    - Spatial and temporal frequency, wavelength, light level, orientation, and surrounding signals (masking).
    - HVS sensitivity changes if observer has ability to zoom, change bands, remap, rotate, and turn on a light.

BASIC STRATEGY IN IMAGE COMPRESSION

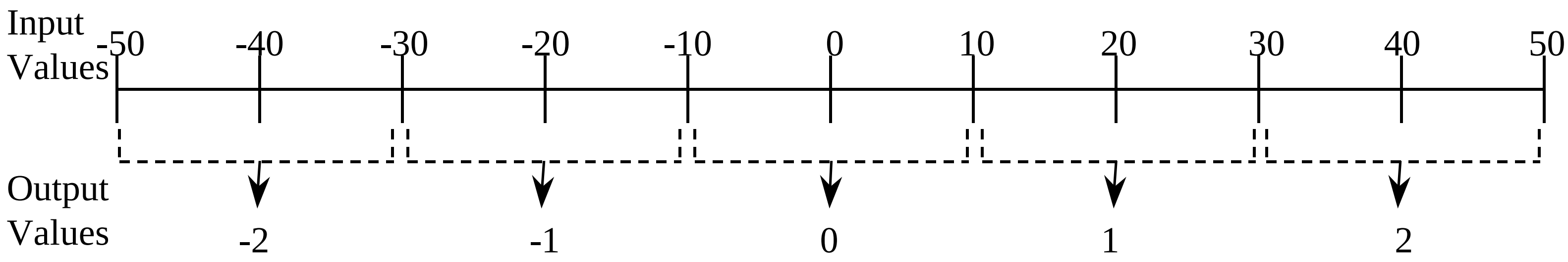
- Ideally, an image compression technique removes redundant and/or irrelevant information, and efficiently encodes what remains.
- Practically, it is often necessary to throw away both non-redundant information and relevant information to achieve the required compression.
- In either case, the trick is finding methods that allow important information to be efficiently extracted and represented.

TYPES OF COMPRESSION

- **Lossless (Reversible)**
  - The reconstructed image is numerically identical to the original image on a pixel-by-pixel basis.
  - Only a modest amount of compression can be achieved (2:1)
- **Lossy (Irreversible)**
  - The reconstructed image contains degradations relative to the original image, and lower bit rates can be achieved by allowing more degradations.
  - Visually lossless is used to describe lossy compression schemes that result in no "visible" degradations (which is highly dependent on the image, display, and observer).

QUANTIZATION

- Quantization is a many-to-one mapping that reduces the number of possible signal values (which reduces bit rate) at the cost of introducing errors (that reduce image quality).
  - It is the key component that distinguishes lossy compression from lossless compression
- A scalar quantizer individually quantizes each signal value.
- Quantization allows us to trade-off image quality for a lower bit rate.
  - Increasing the quantization bin size (a.k.a. step size) introduces more reconstruction errors but allows for greater compression ratio.



SYMBOL ENCODING

- Symbol encoding involves assigning a binary codeword to each possible quantized value
  - Is typically a lossless step, even in lossy compression
- Fixed-Length Codes
  - The number of bits/symbol is constant ⇒ (constant data rate)
  - This is inefficient if the number of symbols is not a power of two, or if the symbol probabilities are not equal
  - Less sensitive to communication errors
- Variable-Length Codes
  - The number of bits/symbol is variable ⇒ (variable data rate)
  - Generally, this is much more efficient than fixed-length coding
  - Variable-length coding techniques include Huffman coding and arithmetic coding.

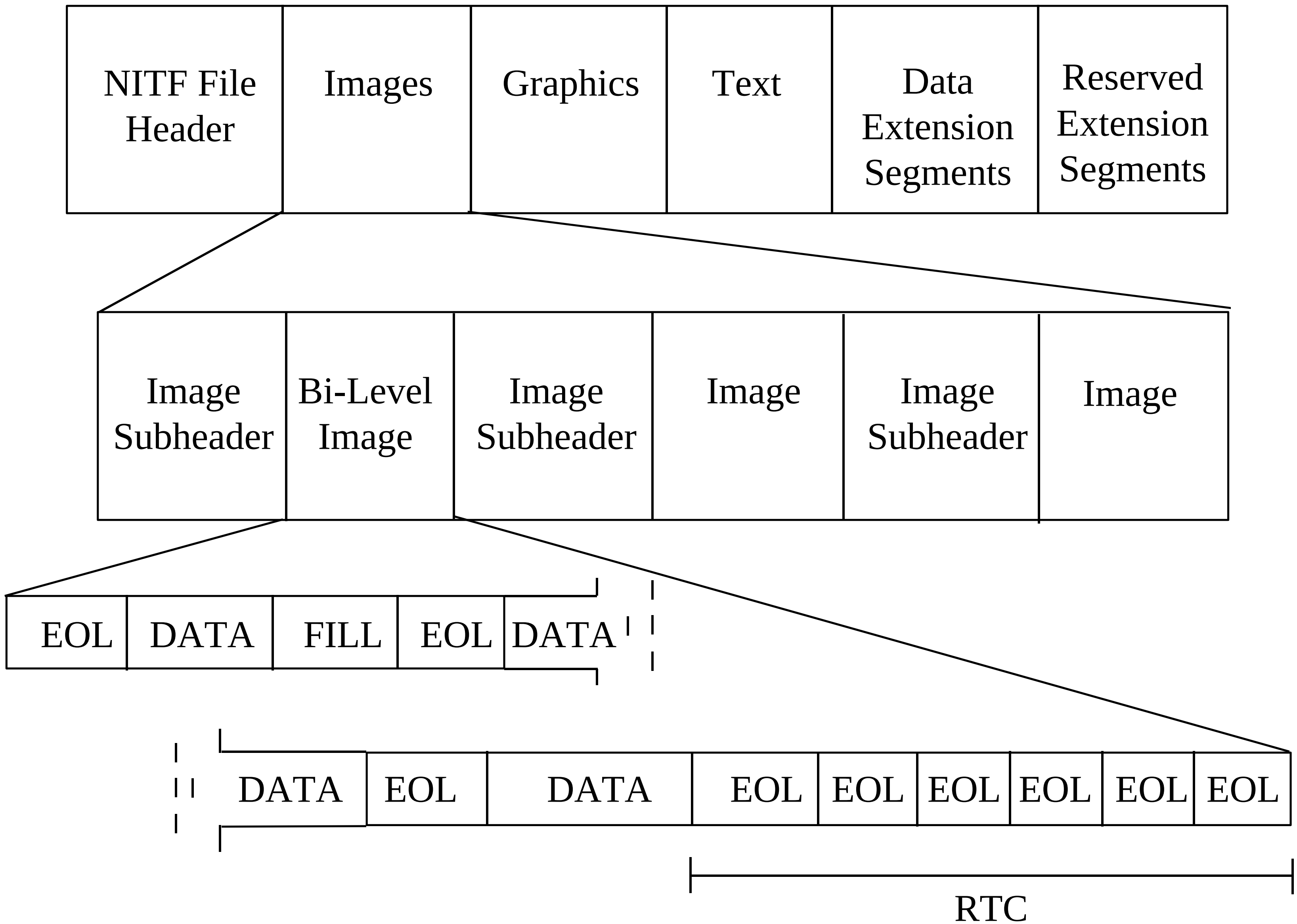
COMPRESSION

BI-LEVEL

MIL-STD-188-196

ITU-T RECMN T.4 AMD2

BI-LEVEL NITF FILE STRUCTURE



ONE DIMENSIONAL ENCODED IMAGE

0000	1000	1111	first line	white = 0
				black = 1
1100	0000	0000	second line	always start w/white run length

The corresponding encoded binary stream for this image is depicted below:

code for 4 white   code for 3 white

EOL   1011   0101   0000   11   EOL

code for 1 black   code for 4 black

code for 2 black

0011   0101   1100   111   EOL   EOL   EOL   EOL   EOL   EOL

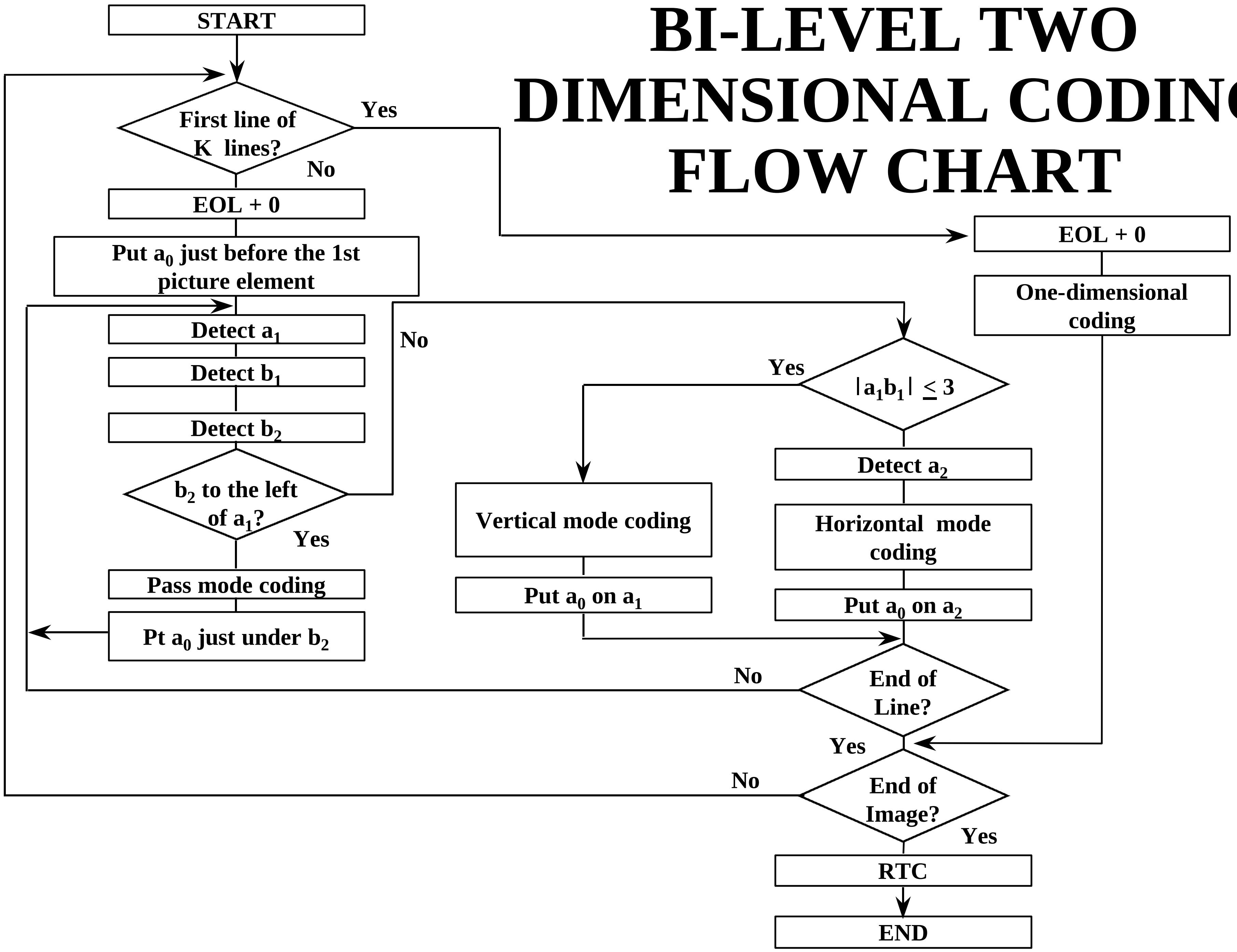
code for 0 white   code for 10 white

Note: ***EOL represents*** ***0000*** ***0000*** ***0001*** ***bit stream***

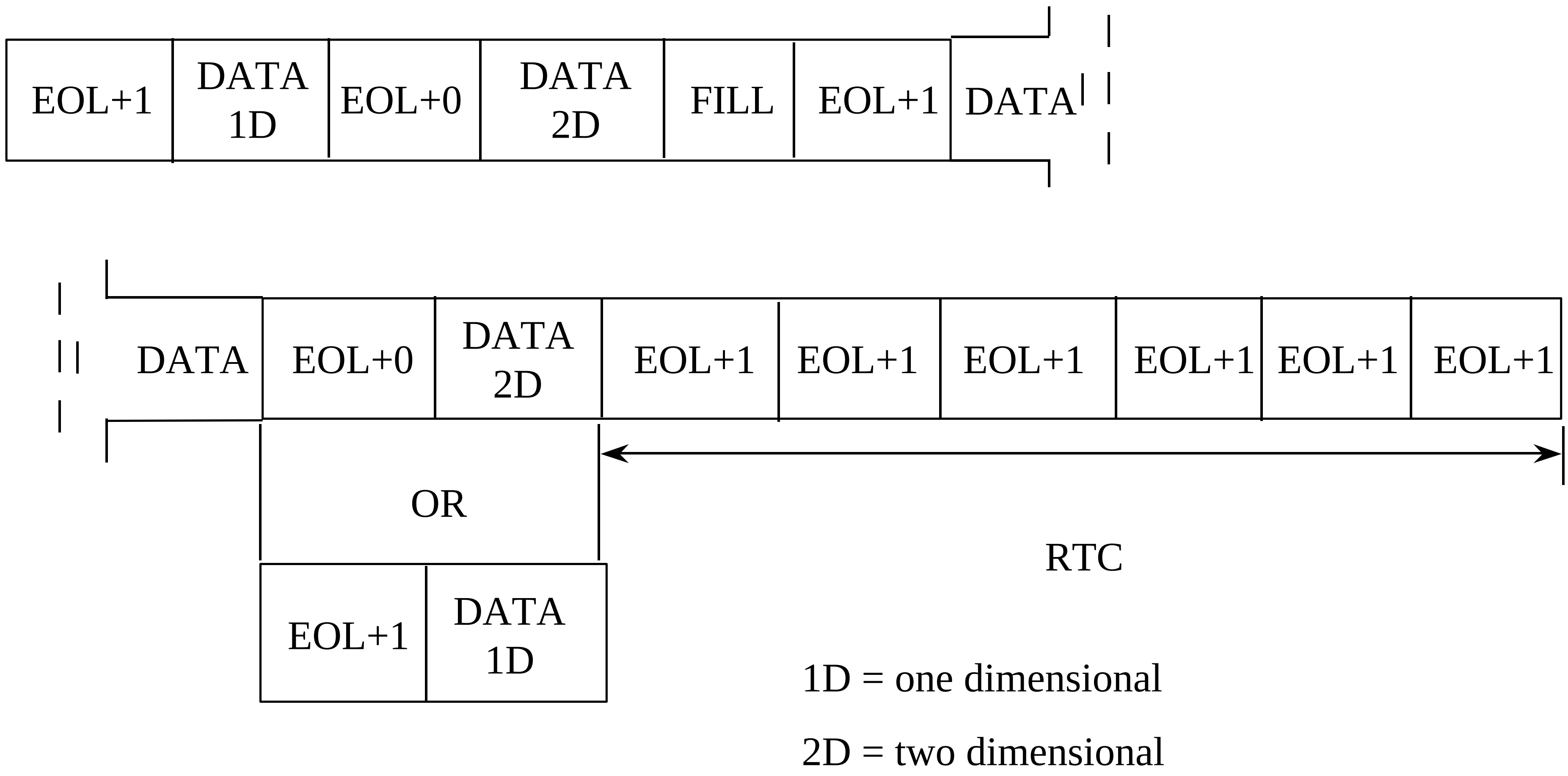
TWO DIMENSIONAL CODES

MODE	ELEMENTS TO BE CODED	NOTATION	CODE WORD
PASS	b_1, b_1	P	0001
HORIZONTAL	a_1 a_1, a_1 a_1	H	$001+M(a_0,a_1)+M(a_1,a_2)$ (Note $_1$)
VERTICAL	a_1 just under b_1	a_1 b_1 = 0 V(0)	1
	a_1 to rt of b_1	a_1 b_1 = 1 $V_R(1)$	011
		a_1 b_1 = 2 $V_R(2)$	000011
		a_1 b_1 = 3 $V_R(3)$	0000011
	a_1 to the left of b_1	a_1 b_1 = 1 $V_L(1)$	010
		a_1 b_1 = 2 $V_L(2)$	000010
		a_1 b_1 = 3 $V_L(3)$	0000010

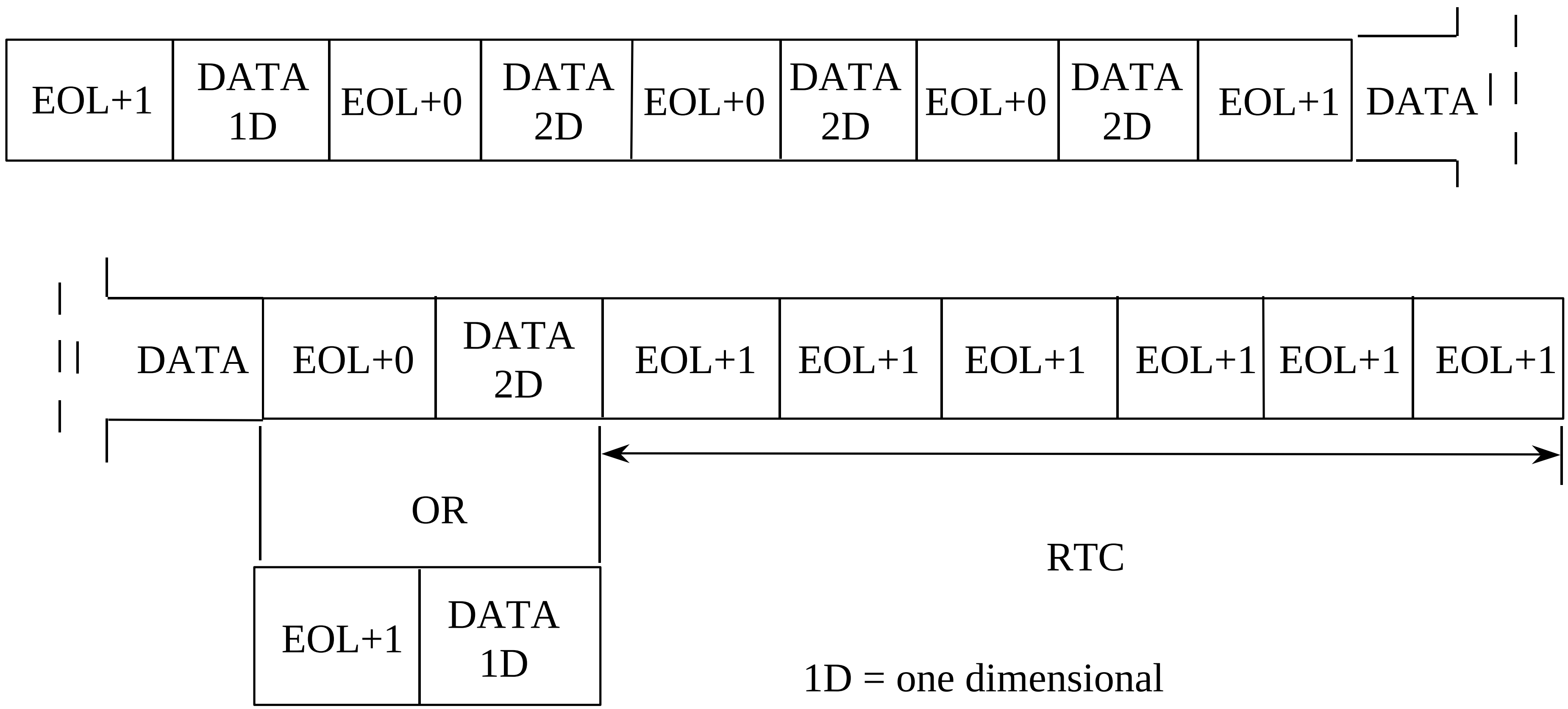
BI-LEVEL TWO DIMENSIONAL CODING FLOW CHART



TWO DIMENSIONAL ENCODED IMAGE DATA ORGANIZATION (K=2)



TWO DIMENSIONAL ENCODED IMAGE DATA ORGANIZATION (K=4)



1D = one dimensional

2D = two dimensional

TWO DIMENSIONAL ENCODING

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
R	0	1	1	0	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1	1	1
C	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	1	1	0	0	0	0	0

The data to be compressed (R = reference line, C = coding line).

First line is encoded one-dimensionally. (EOL +1) 0001 1111 0111 1110 0011 1001 1011 (EOL +0).

The second line is encoded two-dimensionally as follows:

Step 1

Parameter	a0	a1	a2	b1	b2
Position	C0	C2	NA	R2	R4
Mode	Vertical Mode V(0)				
Coded Data	1				

Step 2

Parameter	a0	a1	a2	b1	b2
Position	C2	C3	NA	R4	R8
Mode	Vertical Mode $V_L(1)$				
Coded Data	010				

Step 3

Parameter	a0	a1	a2	b1	b2
Position	C3	C10	NA	R6	R8
Mode	Pass Mode				
Coded Data	0001				

Step 4

Parameter	a0	a1	a2	b1	b2
Position	C8	C10	NA	R11	R13
Mode	Vertical Mode $V_L(1)$				
Coded Data	010				

TWO DIMENSIONAL ENCODING (CONT'D)

Step 5

Parameter	a0	a1	a2	b1	b2
Position	C10	C13	NA	R13	R21
Mode	Vertical Mode V(0)				
Coded Data	1				

Step 6

Parameter	a0	a1	a2	b1	b2
Position	C13	C16	C20	R21	R25
Mode	Horizontal Mode -White Run of 3 -Black Run of 4				
Coded Data	010 1000 011				

Step 7

Parameter	a0	a1	a2	b1	b2
Position	C20	C25	C25	R21	R25
Mode	Horizontal Mode -White Run of 5 -Black Run of 0				
Coded Data	010 1110 0000 1101 11				

Corresponding encoded bit stream is:

(EOL+1) 0001 1111 0111 1110 0011 1001 1011 (EOL+0)

1010 0001 0101 0011 0000 1100 1111 0000 0110 111
(EOL+1)

(EOL+1) (EOL+1) (EOL+1) (EOL+1) (EOL+1)